

---

---

**32-bit General Purpose and Motor Control Application MCUs with  
CAN FD, FPU, ECC Flash, and up to 512 KB Flash, 64 KB SRAM, and  
Op amps**

---

---

**Operating Conditions: 2.3V to 3.6V**

- -40°C to +85°C, DC to 120 MHz
- -40°C to +125°C, DC to 80 MHz

**Core: 120 MHz (up to 198 DMIPS)**

- MIPS32<sup>®</sup> microAptiv™ MCU core with Floating Point Unit
- microMIPS™ mode for up to 40% smaller code size
- DSP-enhanced core:
  - Four 64-bit accumulators
  - Single-cycle MAC, saturating and fractional math
- Code-efficient (C and Assembly) architecture
- Two 32-bit core register files to reduce interrupt latency

**Clock Management**

- 8 MHz ±4% (FRC) internal oscillator -40°C to +85°C
- Programmable PLLs and oscillator clock sources:
  - HS and EC clock modes
- 32 kHz Internal Low-power RC oscillator (LPRC)
- Independent external low-power 32 kHz crystal oscillator
- Fail-Safe Clock Monitor (FSCM)
- Independent Watchdog Timers (WDT) and Deadman Timer (DMT)
- Fast wake-up and start-up
- Four Fractional clock out (REFCLKO) modules

**Power Management**

- Low-power management modes (Sleep, and Idle)
- Integrated:
  - Power-on Reset (POR) and Brown-out Reset (BOR)
  - Programmable High/Low Voltage Detect (HLVD)
- On-board capacitorless regulator

**Motor Control PWM**

- Up to nine PWM pairs
- Leading-edge and Trailing-edge blanking
- Dead Time for rising and falling edges
- Dead Time Compensation
- 8.33 ns PWM Resolution
- Clock Chopping for High-Frequency Operation
- PWM Support for:
  - DC/DC, AC/DC, inverters, PFC, lighting
  - BLDC, PMSM, ACIM, SRM motors
- Choice of 10 Fault and 9 Current Limit Inputs
- Flexible Trigger Configuration for ADC Triggering

**Motor Encoder Interface**

- Three Quadrature Encoder Interface (QE1) modules:
  - Four inputs: Phase A, Phase B, Home, and Index

**Audio/Graphics/Touch Interfaces**

- Up to two I<sup>2</sup>S audio data communication interfaces
- Up to two SPI control interfaces
- Programmable master clock:
  - Generation of fractional clock frequencies
  - Can be tuned in run-time

**Unique Features**

- Permanent non-volatile 4-word unique device serial number
- Flash Error Code Correction (ECC)

**Direct Memory Access (DMA)**

- Up to eight channels with automatic data size detection
- Programmable Cyclic Redundancy Check (CRC)
- Up to 64 KB transfers

**Security Features**

- Advanced Memory Protection:
  - Peripheral and memory region access control

**Advanced Analog Features**

- 12-bit ADC module:
  - Sum of all individual ADCs combined, 25.45 Msps 12-bit mode or 33.79 Msps 8-bit mode
  - 7 individual ADC modules
  - 3.75 Msps per S&H with dedicated DMA
  - Up to 30 analog inputs
- Flexible and independent ADC trigger sources
- Four high bandwidth op-amps and five comparators
- Up to two 12-bit CDACs
- Internal temperature sensor ±2°C accuracy
- Capacitive Touch Divider (CVD)

**Communication Interfaces**

- CAN Flexible Data-Rate (CAN FD) module (with dedicated DMA channels):
  - 2.0B Active with DeviceNet™ addressing support
  - ISO 11898-1:2015 compliant
- Up to two UART modules (up to 25 Mbps):
  - Supports LIN 2.1 and IrDA<sup>®</sup> protocols
- Two SPI/I<sup>2</sup>S modules (SPI 50 Mbps)
- Two I<sup>2</sup>C modules (up to 1 Mbaud) with SMBus support
- Peripheral Pin Select (PPS) to enable remappable pin functions

**Timers/Output Compare/Input Capture/RTCC**

- Up to nine 16-bit or one 16-bit and eight 32-bit timers/counters for GP and MC devices and two additional QE1 32-bit timers for MC devices
- 9 Output Compare (OC) modules
- 9 Input Capture (IC) modules
- PPS to enable function remap
- Real-Time Clock and Calendar (RTCC) module

**Input/Output**

- 5V-tolerant pins with up to 22 mA source/sink
- Selectable internal open drain, pull-ups, and pull-downs
- External interrupts on all I/O pins
- Five programmable edge/level-triggered interrupt pins

**Qualification and Class B Support**

- Class B Safety Library, IEC 60730 (planned)
- Back-up internal oscillator
- Clock monitor with back-up internal oscillator
- Global register locking

**Debugger Development Support**

- In-circuit and in-application programming
- 2-wire or 4-wire MIPS<sup>®</sup> Enhanced JTAG interface
- Unlimited software and 12 complex breakpoints
- IEEE 1149.2-compatible (JTAG) boundary scan
- Non-intrusive hardware-based instruction trace

**Software and Tools Support**

- C/C++ compiler with native DSP/fractional support
- MPLAB<sup>®</sup> Harmony Integrated Software Framework
- TCP/IP, Graphics, and mTouch™ middleware
- MFi, Android™ and Bluetooth<sup>®</sup> audio frameworks
- RTOS Kernels: Express Logic ThreadX, FreeRTOS™, OPENRTOS<sup>®</sup>, Micrium<sup>®</sup> µC/OS™, and SEGGER embOS<sup>®</sup>

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## Packages

| Type               | VQFN           | QFN          | TQFP           |                |
|--------------------|----------------|--------------|----------------|----------------|
| Pin Count          | 48             | 64           | 48             | 64             |
| I/O Pins (up to)   | 37             | 53           | 37             | 53             |
| Contact/Lead Pitch | 0.4 mm         | 0.50 mm      | 0.5 mm         | 0.50 mm        |
| Dimensions         | 6 x 6 x 0.9 mm | 9 x 9 x 1 mm | 7 x 7 x 0.9 mm | 10 x 10 x 1 mm |

**TABLE 1: PIC32MK MOTOR CONTROL AND GENERAL PURPOSE (MC AND GP) FAMILY FEATURES**

| Device            | Program Memory | Data Memory | Floating Point Unit (FPU) | Pins     | Packages     | Boot Flash Memory | Remappable Pin Functions              |      |                      | Motor Control PWM Pairs | QEI (Quadrature Encoder) | CAN FD 2.0B | DMA Channels w/CRC Programmable/Dedicated (Programmable/Dedicated) | Independent ADC Modules | ADC Channels | Op Amp/Comparator | I <sup>2</sup> C | RTCC | REFCLK | CDAC (Control DAC) | CTMU | CLC (Configurable Logic Cell) | HLVD | I/O Pins | JTAG/ICSP | Trace |                                    |
|-------------------|----------------|-------------|---------------------------|----------|--------------|-------------------|---------------------------------------|------|----------------------|-------------------------|--------------------------|-------------|--|-------------------------|--------------|-------------------|------------------|------|--------|--------------------|------|-------------------------------|------|----------|-----------|-------|------------------------------------|
|                   |                |             |                           |          |              |                   | Timers/Capture/Compare <sup>(1)</sup> | UART | SPI/I <sup>2</sup> S |                         |                          |             |  |                         |              |                   |                  |      |        |                    |      |                               |      |          |           |       | External Interrupts <sup>(2)</sup> |
| PIC32MK0512MCJ064 | 512K           | 64K         | Y                         | 64<br>64 | TQFP<br>UQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 9                        | 3           | 1  | 8/2                     | 7            | 30                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 53        | Y     | Y                                  |
| PIC32MK0512MCJ048 | 512K           | 64K         | Y                         | 48<br>48 | TQFP<br>VQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 6                        | 3           | 1  | 8/2                     | 7            | 18                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 37        | Y     | N                                  |
| PIC32MK0256MCJ064 | 256K           | 64K         | Y                         | 64<br>64 | TQFP<br>UQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 9                        | 3           | 1  | 8/2                     | 7            | 30                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 53        | Y     | Y                                  |
| PIC32MK0256MCJ048 | 256K           | 64K         | Y                         | 48<br>48 | TQFP<br>VQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 6                        | 3           | 1  | 8/2                     | 7            | 18                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 37        | Y     | N                                  |
| PIC32MK0512GPG064 | 512K           | 64K         | Y                         | 64<br>64 | TQFP<br>UQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 0                        | 0           | 0  | 8/1                     | 7            | 30                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 53        | Y     | Y                                  |
| PIC32MK0512GPG048 | 512K           | 64K         | Y                         | 48<br>48 | TQFP<br>VQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 0                        | 0           | 0  | 8/1                     | 7            | 18                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 37        | Y     | N                                  |
| PIC32MK0256GPG064 | 256K           | 64K         | Y                         | 64<br>64 | TQFP<br>UQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 0                        | 0           | 0  | 8/1                     | 7            | 30                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 53        | Y     | Y                                  |
| PIC32MK0256GPG048 | 256K           | 64K         | Y                         | 48<br>48 | TQFP<br>VQFN | 16                | 9/9/9                                 | 2    | 2/2                  | 5                       | 0                        | 0           | 0  | 8/1                     | 7            | 18                | 4/5              | 2    | Y      | 4                  | 2    | 1                             | 4    | 1        | 37        | Y     | N                                  |

**Note** 1: Eight out of nine timers are remappable.  
2: Four out of five external interrupts are remappable.

# PIC32MK GPG/MCJ with CAN FD Family

## Device Pin Tables

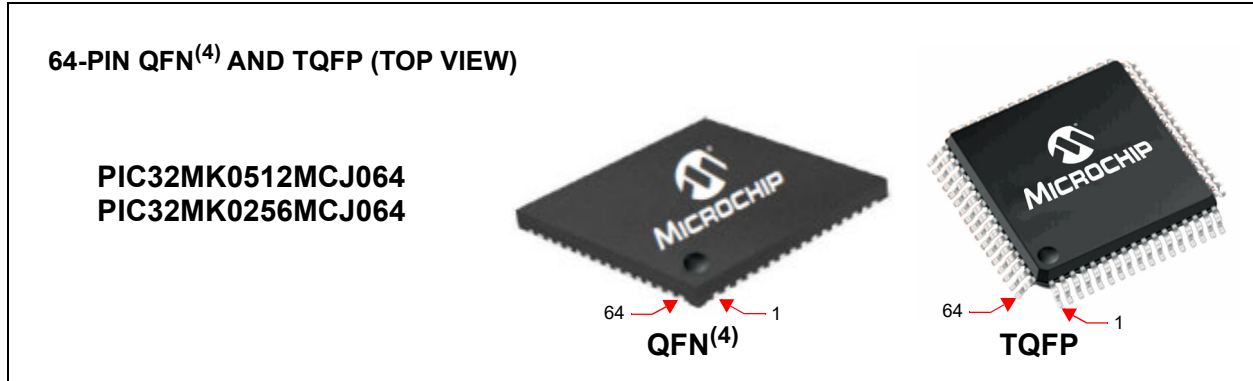
TABLE 2: PIN NAMES FOR 64-PIN GENERAL PURPOSE (GPG) DEVICES

| 64-PIN QFN <sup>(4)</sup> AND TQFP (TOP VIEW)  |  |       |  |
|--|--|-------|--|
|   |  |       |  |
| <p>PIC32MK0512GPG064<br/>PIC32MK0256GPG064</p>   |  |       |  |
| <p>64 → 1      QFN<sup>(4)</sup>      64 → 1      TQFP</p>   |  |       |  |
| Pin #  | Full Pin Name                                      | Pin # | Full Pin Name  |
| 1  | TCK/RPA7/RA7                                       | 33    | OA5IN+/AN24/CVD24/C5IN1+/C5IN3-/RPA4/RA4                           |
| 2  | RPB14/RB14   | 34    | AN40/CVD40/RPE0/RE0  |
| 3  | RPB15/RB15   | 35    | AN41/CVD41/RPE1/RE1  |
| 4  | AN19/CVD19/RPG6/RG6                                | 36    | AN46/CVD46/RPA14/RA14  |
| 5  | AN18/CVD18/RPG7/RG7                                | 37    | AN47/CVD47/RPA15/RA15  |
| 6  | AN17/CVD17/RPG8/RG8                                | 38    | VDD  |
| 7  | MCLR#  | 39    | OSCI/CLKI/AN49/CVD49/PC12/RC12                                     |
| 8  | AN16/CVD16/RPG9/RG9                                | 40    | OSCO/CLKO/PC15/RC15  |
| 9  | VSS  | 41    | VSS  |
| 10   | VDD  | 42    | RD8  |
| 11   | AN10/CVD10/RPA12/RA12                              | 43    | PGD2/RPB5/SDA1/RB5   |
| 12   | AN9/CVD9/RPA11/RA11                                | 44    | PGC2/RPB6/SCL1/RB6   |
| 13   | OA2OUT/AN0/C2IN4-/C4IN3-/RPA0/RA0                  | 45    | DAC1/AN48/CVD48/PC10/RC10  |
| 14   | OA2IN+/AN1/C2IN1+/RPA1/RA1                         | 46    | OA5OUT/AN25/CVD25/C5IN4-/RPB7/SCK1/INT0/RB7                        |
| 15   | PGD3/VREF-/OA2IN-/AN2/C2IN1-/RPB0/CTED2/RB0        | 47    | SOSCI/PC13 <sup>(5)</sup> /RC13 <sup>(5)</sup>                     |
| 16   | PGC3/OA1OUT/VREF+/AN3/C1IN4-/C4IN2-/RPB1/CTED1/RB1 | 48    | SOSCO/RPB8 <sup>(5)</sup> /T1CK <sup>(5)</sup> /RB8 <sup>(5)</sup> |
| 17   | PGC1/OA1IN+/AN4/C1IN1+/C1IN3-/C2IN3-/RPB2/RB2      | 49    | TMS/OA5IN-/AN27/CVD27/LVDIN/C5IN1-/RPB9/RB9                        |
| 18   | PGD1/OA1IN-/AN5/CTCMP/C1IN1-/RTCC/RPB3/RB3         | 50    | TRCLK/PC6/RC6  |
| 19   | AVDD   | 51    | TRD0/PC7/RC7   |
| 20   | AVSS   | 52    | TRD1/PC8/RC8   |
| 21   | OA3OUT/AN6/CVD6/C3IN4-/C4IN1+/C4IN4-/RPC0/RC0      | 53    | TRD2/RPD5/RD5  |
| 22   | OA3IN-/AN7/CVD7/C3IN1-/C4IN1-/RPC1/RC1             | 54    | TRD3/RPD6/RD6  |
| 23   | OA3IN+/AN8/CVD8/C3IN1+/C3IN3-/RPC2/RC2             | 55    | RPC9/RC9   |
| 24   | AN11/CVD11/C1IN2-/RC11                             | 56    | VSS  |
| 25   | VSS  | 57    | VDD  |
| 26   | VDD  | 58    | RPF0/RF0   |
| 27   | AN12/CVD12/C2IN2-/C5IN2-/RE12                      | 59    | RPF1/RF1   |
| 28   | AN13/CVD13/C3IN2-/RE13                             | 60    | RPB10/RB10   |
| 29   | AN14/CVD14/RPE14/RE14                              | 61    | RPB11/RB11   |
| 30   | AN15/CVD15/RPE15/RE15                              | 62    | RPB12/RB12   |
| 31   | TDI/DAC2/AN26/CVD26/RPA8/SDA2/RA8                  | 63    | RPB13/CTPLS/RB13   |
| 32   | RPB4/SCL2/RB4                                      | 64    | TDO/RA10   |

- Note**
- 1: The RPN pins can be used by remappable peripherals. See [Table 1](#) for the available peripherals and **10.3 “Peripheral Pin Select (PPS)”** for restrictions.
  - 2: Every I/O port pin (RAX-RGx) can be used as a change notification pin (CNAX-CNGx). See **10.0 “I/O Ports”** for more information.
  - 3: Shaded pins are 5V tolerant.
  - 4: The metal heat sink pad at the bottom of the device is not connected to any pins and is recommended to be connected to Vss externally.
  - 5: Functions are restricted to input functions only and inputs will be slower than the standard inputs. Change notification interrupt is not available on this pin.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 3: PIN NAMES FOR 64-PIN MOTOR CONTROL (MCJ) DEVICES**



| Pin # | Full Pin Name                                      | Pin # | Full Pin Name  |
|-------|--|-------|--|
| 1     | TCK/RPA7/PWM4L/RA7                                 | 33    | OA5IN+/AN24/CVD24/C5IN1+/C5IN3-/RPA4/RA4                           |
| 2     | RPB14/PWM1H/RB14                                   | 34    | AN40/CVD40/RPE0/RE0  |
| 3     | RPB15/PWM1L/RB15                                   | 35    | AN41/CVD41/RPE1/RE1  |
| 4     | AN19/CVD19/RPG6/PWM7L/RG6                          | 36    | AN46/CVD46/RPA14/RA14  |
| 5     | AN18/CVD18/RPG7/PWM7H/RG7                          | 37    | AN47/CVD47/RPA15/RA15  |
| 6     | AN17/CVD17/RPG8/RG8                                | 38    | VDD  |
| 7     | MCLR#  | 39    | OSCI/CLKI/AN49/CVD49/RPC12/RC12                                    |
| 8     | AN16/CVD16/RPG9/FLT12/RG9                          | 40    | OSCO/CLKO/RPC15/RC15   |
| 9     | VSS  | 41    | VSS  |
| 10    | VDD  | 42    | RD8  |
| 11    | AN10/CVD10/RPA12/FLT13/RA12                        | 43    | PGD2/RPB5/SDA1/RB5   |
| 12    | AN9/CVD9/RPA11/FLT14/RA11                          | 44    | PGC2/RPB6/SCL1/RB6   |
| 13    | OA2OUT/AN0/C2IN4-/C4IN3-/RPA0/RA0                  | 45    | DAC1/AN48/CVD48/RPC10/RC10   |
| 14    | OA2IN+/AN1/C2IN1+/RPA1/RA1                         | 46    | OA5OUT/AN25/CVD25/C5IN4-/RPB7/SCK1/INT0/RB7                        |
| 15    | PGD3/VREF-/OA2IN-/AN2/C2IN1-/RPB0/CTED2/RB0        | 47    | SOSCI/RPC13 <sup>(5)</sup> /RC13 <sup>(5)</sup>                    |
| 16    | PGC3/OA1OUT/VREF+/AN3/C1IN4-/C4IN2-/RPB1/CTED1/RB1 | 48    | SOSCO/RPB8 <sup>(5)</sup> /T1CK <sup>(5)</sup> /RB8 <sup>(5)</sup> |
| 17    | PGC1/OA1IN+/AN4/C1IN1+/C1IN3-/C2IN3-/RPB2/RB2      | 49    | TMS/OA5IN-/AN27/CVD27/LVDIN/C5IN1-/RPB9/RB9                        |
| 18    | PGD1/OA1IN-/AN5/CTCMP/C1IN1-/RTCC/RPB3/RB3         | 50    | TRCLK/RPC6/PWM6H/RC6   |
| 19    | AVDD   | 51    | TRD0/RPC7/PWM6L/RC7  |
| 20    | AVSS   | 52    | TRD1/RPC8/PWM5H/RC8  |
| 21    | OA3OUT/AN6/CVD6/C3IN4-/C4IN1+/C4IN4-/RPC0/RC0      | 53    | TRD2/RPD5/PWM9H/RD5  |
| 22    | OA3IN-/AN7/CVD7/C3IN1-/C4IN1-/RPC1/RC1             | 54    | TRD3/RPD6/PWM9L/RD6  |
| 23    | OA3IN+/AN8/CVD8/C3IN1+/C3IN3-/RPC2/FLT3/RC2        | 55    | RPC9/PWM5L/RC9   |
| 24    | AN11/CVD11/C1IN2-/FLT4/RC11                        | 56    | VSS  |
| 25    | VSS  | 57    | VDD  |
| 26    | VDD  | 58    | RPF0/PWM8H/RF0   |
| 27    | AN12/CVD12/C2IN2-/C5IN2-/FLT5/RE12                 | 59    | RPF1/PWM8L/RF1   |
| 28    | AN13/CVD13/C3IN2-/FLT6/RE13                        | 60    | RPB10/PWM3H/RB10   |
| 29    | AN14/CVD14/RPE14/FLT7/RE14                         | 61    | RPB11/PWM3L/RB11   |
| 30    | AN15/CVD15/RPE15/FLT8/RE15                         | 62    | RPB12/PWM2H/RB12   |
| 31    | TDI/DAC2/AN26/CVD26/RPA8/SDA2/RA8                  | 63    | RPB13/PWM2L/CTPLS/RB13   |
| 32    | FLT15/RPB4/SCL2/RB4                                | 64    | TDO/PWM4H/RA10   |

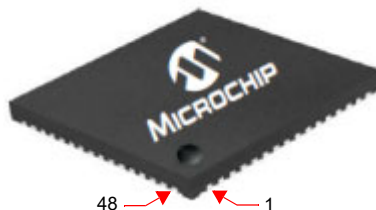
- Note**
- 1: The RPN pins can be used by remappable peripherals. See [Table 1](#) for the available peripherals and **10.3 “Peripheral Pin Select (PPS)”** for restrictions.
  - 2: Every I/O port pin (RAX-RGx) can be used as a change notification pin. See **10.0 “I/O Ports”** for more information.
  - 3: Shaded pins are 5V tolerant.
  - 4: The metal heat sink pad on the bottom of the QFN device is not connected to any pins and is recommended to be connected to VSS externally.
  - 5: Functions are restricted to input functions only and inputs will be slower than standard inputs. Change notification interrupt is not available on this pin.

# PIC32MK GPG/MCJ with CAN FD Family

TABLE 4: PIN NAMES FOR 48-PIN GENERAL PURPOSE (GPG) DEVICES

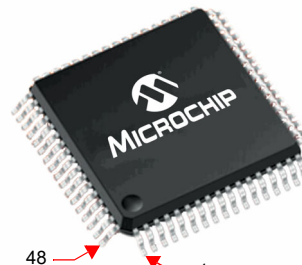
## 48-PIN VQFN<sup>(4)</sup> AND TQFP (TOP VIEW)

PIC32MK0512GPG048  
PIC32MK0256GPG048



48 1

VQFN (4)



48 1

TQFP

| Pin # | Full Pin Name  | Pin # | Full Pin Name  |
|-------|--|-------|--|
| 1     | TCK / RPA7 / RA7   | 25    | OA5IN+ / AN24 / CVD24 / C5IN1+ / C5IN3- / RPA4 / RA4               |
| 2     | RPB14 / RB14   | 26    | VDD  |
| 3     | RPB15 / RB15   | 27    | OSCI/CLKI / AN49 / CVD49 / RPC12 / RC12                            |
| 4     | MCLR#  | 28    | OSCO / CLKO / RPC15 / RC15   |
| 5     | VSS  | 29    | VSS  |
| 6     | VDD  | 30    | RD8  |
| 7     | AN10 / CVD10 / RPA12 / RA12  | 31    | PGD2 / RPB5 / SDA1 / RB5   |
| 8     | AN9 / CVD9 / RPA11 / RA11  | 32    | PGC2 / RPB6 / SCL1 / RB6   |
| 9     | OA2OUT / AN0 / C2IN4- / C4IN3- / RPA0 / RA0                        | 33    | DAC1 / AN48 / CVD48 / RPC10 / RC10                                 |
| 10    | OA2IN+ / AN1 / C2IN1+ / RPA1 / RA1                                 | 34    | OA5OUT / AN25 / CVD25 / C5IN4- / RPB7 / SCK1 / INT0 / RB7          |
| 11    | PGD3 / VREF- / OA2IN- / AN2 / C2IN1- / RPB0 / CTED2 / RB0          | 35    | SOSCI/RPC13 <sup>(5)</sup> /RC13 <sup>(5)</sup>                    |
| 12    | PGC3 / OA1OUT / VREF+ / AN3 / C1IN4- / C4IN2- / RPB1 / CTED1 / RB1 | 36    | SOSCO/RPB8 <sup>(5)</sup> /T1CK <sup>(5)</sup> /RB8 <sup>(5)</sup> |
| 13    | PGC1 / OA1IN+ / AN4 / C1IN1+ / C1IN3- / C2IN3- / RPB2 / RB2        | 37    | TMS / OA5IN- / AN27 / CVD27 / LVDIN / C5IN1- / RPB9 / RB9          |
| 14    | PGD1 / OA1IN- / AN5 / CTCMP / C1IN1- / RTCC / RPB3 / RB3           | 38    | RPC6 / RC6   |
| 15    | AVDD   | 39    | RPC7 / RC7   |
| 16    | AVSS   | 40    | RPC8 / RC8   |
| 17    | OA3OUT / AN6 / CVD6 / C3IN4- / C4IN1+ / C4IN4- / RPC0 / RC0        | 41    | RPC9 / RC9   |
| 18    | OA3IN- / AN7 / CVD7 / C3IN1- / C4IN1- / RPC1 / RC1                 | 42    | VSS  |
| 19    | OA3IN+ / AN8 / CVD8 / C3IN1+ / C3IN3- / RPC2 / RC2                 | 43    | VDD  |
| 20    | AN11 / CVD11 / C1IN2- / RC11                                       | 44    | RPB10 / RB10   |
| 21    | VSS  | 45    | RPB11 / RB11   |
| 22    | VDD  | 46    | RPB12 / RB12   |
| 23    | TDI / DAC2 / AN26 / CVD26 / RPA8 / SDA2 / RA8                      | 47    | RPB13 / CTPLS / RB13   |
| 24    | RPB4 / SCL2 / RB4  | 48    | TDO / RA10   |

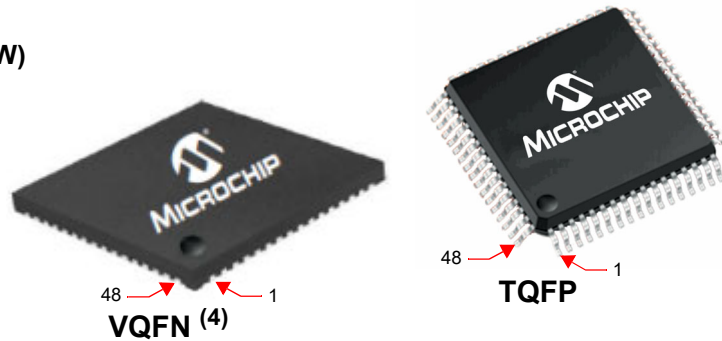
- Note**
- 1: The RPN pins can be used by remappable peripherals. See [Table 1](#) for the available peripherals and [10.3 “Peripheral Pin Select \(PPS\)”](#) for restrictions.
  - 2: Every I/O port pin (RAX-RGx) can be used as a change notification interrupt pin. See [10.0 “I/O Ports”](#) for more information.
  - 3: Shaded pins are 5V tolerant.
  - 4: The metal heat sink pad on the bottom of the QFN device is not connected to any pins and is recommended to be connected to VSS externally.
  - 5: Functions are restricted to input functions only and inputs will be slower than standard inputs. Change notification interrupt is not available on this pin.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 5: PIN NAMES FOR 48-PIN MOTOR CONTROL (MCJ) DEVICES**

**48-PIN VQFN<sup>(4)</sup> AND TQFP (TOP VIEW)**

**PIC32MK0512MCJ048**  
**PIC32MK0256MCJ048**



| Pin # | Full Pin Name  | Pin # | Full Pin Name  |
|-------|--|-------|--|
| 1     | TCK / RPA7 / PWM4L / RA7   | 25    | OA5IN+ / AN24 / CVD24 / C5IN1+ / C5IN3- / RPA4 / RA4                   |
| 2     | RPB14 / PWM1H / RB14   | 26    | VDD  |
| 3     | RPB15 / PWM1L / RB15   | 27    | OSCI / CLKI / AN49 / CVD49 / RPC12 / RC12'                             |
| 4     | MCLR#  | 28    | OSCO / CLKO / RPC15 / RC15   |
| 5     | VSS  | 29    | VSS  |
| 6     | VDD  | 30    | RD8  |
| 7     | AN10 / CVD10 / RPA12 / FLT13 / RA12                                | 31    | PGD2 / RPB5 / SDA1 / RB5   |
| 8     | AN9 / CVD9 / RPA11 / FLT14 / RA11                                  | 32    | PGC2 / RPB6 / SCL1 / RB6   |
| 9     | OA2OUT / AN0 / C2IN4- / C4IN3- / RPA0 / RA0                        | 33    | DAC1 / AN48 / CVD48 / RPC10 / RC10                                     |
| 10    | OA2IN+ / AN1 / C2IN1+ / RPA1 / RA1                                 | 34    | OA5OUT / AN25 / CVD25 / C5IN4- / RPB7 / SCK1 / INT0 / RB7              |
| 11    | PGD3 / VREF- / OA2IN- / AN2 / C2IN1- / RPB0 / CTED2 / RB0          | 35    | SOSCI / RPC13 <sup>(5)</sup> / RC13 <sup>(5)</sup>                     |
| 12    | PGC3 / OA1OUT / VREF+ / AN3 / C1IN4- / C4IN2- / RPB1 / CTED1 / RB1 | 36    | SOSCO / RPB8 <sup>(5)</sup> / T1CK <sup>(5)</sup> / RB8 <sup>(5)</sup> |
| 13    | PGC1 / OA1IN+ / AN4 / C1IN1+ / C1IN3- / C2IN3- / RPB2 / RB2        | 37    | TMS / OA5IN- / AN27 / CVD27 / LVDIN / C5IN1- / RPB9 / RB9              |
| 14    | PGD1 / OA1IN- / AN5 / CTCMP / C1IN1- / RTCC / RPB3 / RB3           | 38    | RPC6 / PWM6H / RC6   |
| 15    | AVDD   | 39    | RPC7 / PWM6L / RC7   |
| 16    | AVSS   | 40    | RPC8 / PWM5H / RC8   |
| 17    | OA3OUT / AN6 / CVD6 / C3IN4- / C4IN1+ / C4IN4- / RPC0 / RC0        | 41    | RPC9 / PWM5L / RC9   |
| 18    | OA3IN- / AN7 / CVD7 / C3IN1- / C4IN1- / RPC1 / RC1                 | 42    | VSS  |
| 19    | OA3IN+ / AN8 / CVD8 / C3IN1+ / C3IN3- / RPC2 / FLT3 / RC2          | 43    | VDD  |
| 20    | AN11 / CVD11 / C1IN2- / FLT4 / RC11                                | 44    | RPB10 / PWM3H / RB10   |
| 21    | VSS  | 45    | RPB11 / PWM3L / RB11   |
| 22    | VDD  | 46    | RPB12 / PWM2H / RB12   |
| 23    | TDI / DAC2 / AN26 / CVD26 / RPA8 / SDA2 / RA8                      | 47    | RPB13 / PWM2L / CTPLS / RB13   |
| 24    | FLT15 / RPB4 / SCL2 / RB4  | 48    | TDO / PWM4H / RA10   |

- Note**
- 1: The RPN pins can be used by remappable peripherals. See [Table 1](#) for the available peripherals and **10.3 “Peripheral Pin Select (PPS)”** for restrictions.
  - 2: Every I/O port pin (RAX-RGx) can be used as a change notification interrupt pin. See **10.0 “I/O Ports”** for more information.
  - 3: Shaded pins are 5V tolerant.
  - 4: The metal heat sink pad on the bottom of the QFN device is not connected to any pins and is recommended to be connected to VSS externally.
  - 5: Functions are restricted to input functions only and inputs will be slower than standard inputs. Change notification interrupt is not available on this pin.

# PIC32MK GPG/MCJ with CAN FD Family

---

---



# PIC32MK GPG/MCJ with CAN FD Family

---

---

## Table of Contents

|      |  |     |
|------|--|-----|
| 1.0  | Device Overview .....  | 8   |
| 2.0  | Guidelines for Getting Started with 32-bit MCUs.....   | 26  |
| 3.0  | CPU.....   | 37  |
| 4.0  | Memory Organization.....   | 59  |
| 5.0  | Flash Program Memory.....  | 81  |
| 6.0  | Resets .....   | 92  |
| 7.0  | CPU Exceptions and Interrupt Controller .....  | 100 |
| 8.0  | Oscillator Configuration .....   | 140 |
| 9.0  | Prefetch Module .....  | 158 |
| 10.0 | Direct Memory Access (DMA) Controller .....  | 164 |
| 11.0 | I/O Ports .....  | 192 |
| 12.0 | Timer1 .....   | 224 |
| 13.0 | Timer2 Through Timer9.....   | 230 |
| 14.0 | Deadman Timer (DMT) .....  | 236 |
| 15.0 | Watchdog Timer (WDT) .....   | 244 |
| 16.0 | Input Capture.....   | 248 |
| 17.0 | Output Compare .....   | 253 |
| 18.0 | Serial Peripheral Interface (SPI) and Inter-IC Sound (I <sup>2</sup> S).....                     | 257 |
| 19.0 | Inter-Integrated Circuit (I <sup>2</sup> C).....   | 267 |
| 20.0 | Universal Asynchronous Receiver Transmitter (UART) .....   | 275 |
| 21.0 | Configurable Logic Cell (CLC).....   | 287 |
| 22.0 | Real-Time Clock and Calendar (RTCC).....   | 301 |
| 23.0 | 12-bit High-Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC)..... | 311 |
| 24.0 | Controller Area Network with Flexible Data-rate (CAN FD) .....                                   | 388 |
| 25.0 | Op Amp/Comparator Module .....   | 451 |
| 26.0 | Charge Time Measurement Unit (CTMU) .....  | 480 |
| 27.0 | Control Digital-to-Analog Converter (CDAC).....  | 486 |
| 28.0 | Quadrature Encoder Interface (QEI) .....   | 489 |
| 29.0 | Motor Control PWM Module .....   | 505 |
| 30.0 | High/Low-Voltage Detect (HLVD).....  | 554 |
| 31.0 | Power-Saving Features .....  | 558 |
| 32.0 | Special Features .....   | 565 |
| 33.0 | Instruction Set .....  | 588 |
| 34.0 | Migration Guide .....  | 586 |
| 35.0 | Development Support.....   | 591 |
| 36.0 | Electrical Characteristics .....   | 595 |
| 37.0 | AC and DC Characteristics Graphs.....  | 652 |
| 38.0 | Packaging Information.....   | 654 |
|      | Product Identification System .....  | 667 |
|      | The Microchip Web Site .....   | 674 |
|      | Customer Change Notification Service .....   | 674 |
|      | Customer Support.....  | 674 |

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at [docerrors@microchip.com](mailto:docerrors@microchip.com). We welcome your feedback.

### Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

### Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

### Customer Notification System

Register on our web site at [www.microchip.com](http://www.microchip.com) to receive the most current information on all of our products.

# PIC32MK GPG/MCJ with CAN FD Family

---

## 1.0 DEVICE OVERVIEW

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the documents listed in the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

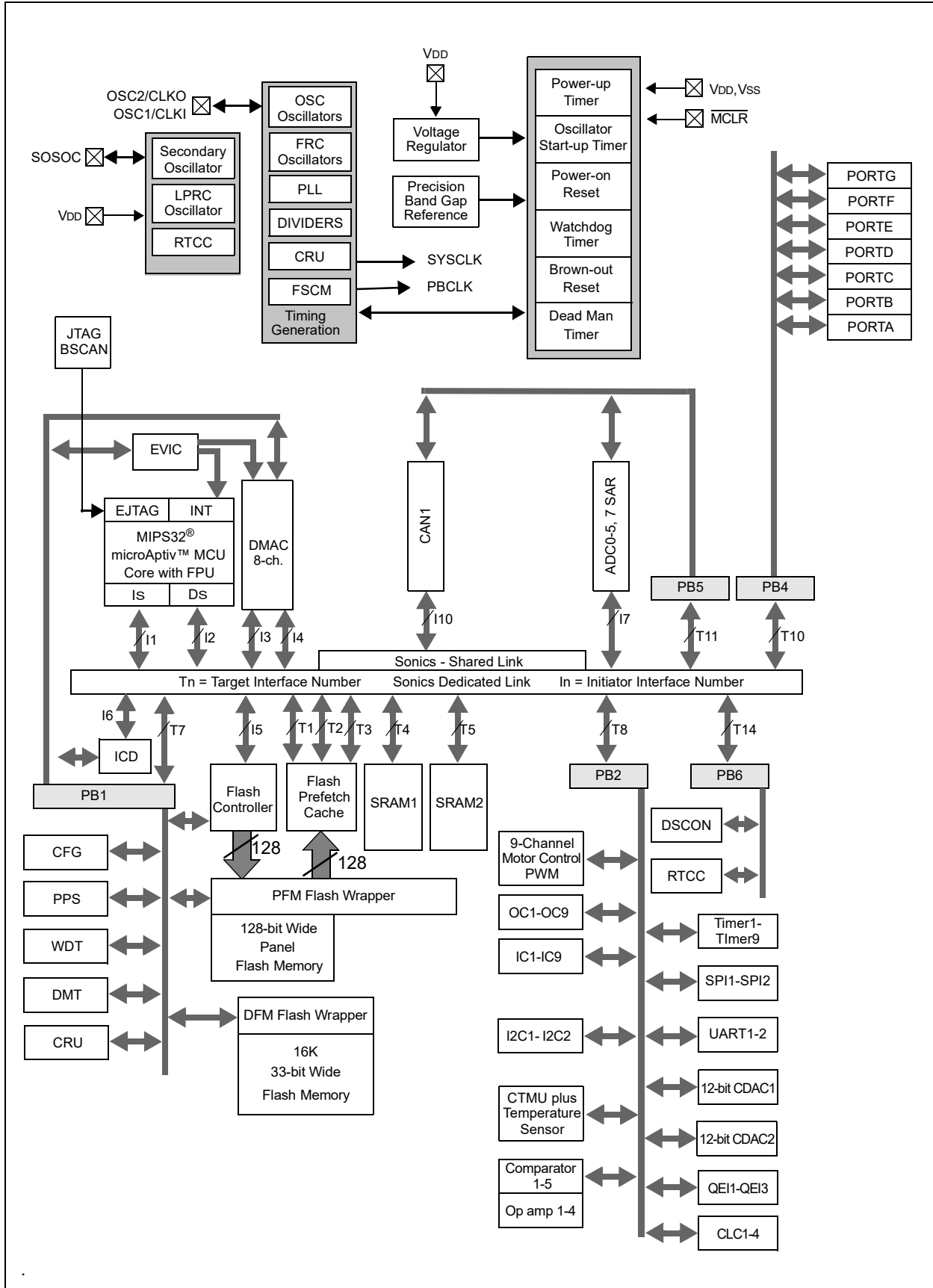
This data sheet contains device-specific information for PIC32MK GPG/MCJ with CAN FD Family of devices.

[Figure 1-1](#) illustrates a general block diagram of the core and peripheral modules in the PIC32MK GPG/MCJ with CAN FD Family of devices.

[Table 1-21](#) through [Table 1-22](#) list the pinout I/O descriptions for the pins shown in the device pin tables (see [Table 2](#) and [Table 4](#)).

# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 1-1: PIC32MK GPG/MCJ WITH CAN FD FAMILY BLOCK DIAGRAM



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-1: PORTA THROUGH PORTG REMAPPABLE PERIPHERAL DESCRIPTIONS**

| Pin Name      | Pin Number      |                  | Pin Type | Buffer Type | Description                           |
|---------------|-----------------|------------------|----------|-------------|---------------------------------------|
|               | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                       |
| <b>PORT A</b> |                 |                  |          |             |                                       |
| RPA0          | 13              | 9                | I/O      | ST          | Remappable PORTA Peripheral Functions |
| RPA1          | 14              | 10               | I/O      | ST          |                                       |
| RPA4          | 33              | 25               | I/O      | ST          |                                       |
| RPA7          | 1               | 1                | I/O      | ST          |                                       |
| RPA8          | 31              | 23               | I/O      | ST          |                                       |
| RPA11         | 12              | 8                | I/O      | ST          |                                       |
| RPA12         | 11              | 7                | I/O      | ST          |                                       |
| RPA14         | 36              | —                | I/O      | ST          |                                       |
| RPA15         | 37              | —                | I/O      | ST          |                                       |
| <b>PORT B</b> |                 |                  |          |             |                                       |
| RPB0          | 15              | 11               | I/O      | ST          | Remappable PORTB Peripheral Functions |
| RPB1          | 16              | 12               | I/O      | ST          |                                       |
| RPB2          | 17              | 13               | I/O      | ST          |                                       |
| RPB3          | 18              | 14               | I/O      | ST          |                                       |
| RPB4          | 32              | 24               | I/O      | ST          |                                       |
| RPB5          | 43              | 31               | I/O      | ST          |                                       |
| RPB6          | 44              | 32               | I/O      | ST          |                                       |
| RPB7          | 46              | 34               | I/O      | ST          |                                       |
| RPB8          | 48              | 36               | I/O      | ST          |                                       |
| RPB9          | 49              | 37               | I/O      | ST          |                                       |
| RPB10         | 60              | 44               | I/O      | ST          |                                       |
| RPB11         | 61              | 45               | I/O      | ST          |                                       |
| RPB12         | 62              | 46               | I/O      | ST          |                                       |
| RPB13         | 63              | 47               | I/O      | ST          |                                       |
| RPB14         | 2               | 2                | I/O      | ST          |                                       |
| RPB15         | 3               | 3                | I/O      | ST          |                                       |
| <b>PORT C</b> |                 |                  |          |             |                                       |
| RPC0          | 21              | 17               | I/O      | ST          | Remappable PORTC Peripheral Functions |
| RPC1          | 22              | 18               | I/O      | ST          |                                       |
| RPC2          | 23              | 19               | I/O      | ST          |                                       |
| RPC6          | 50              | 38               | I/O      | ST          |                                       |
| RPC7          | 51              | 39               | I/O      | ST          |                                       |
| RPC8          | 52              | 40               | I/O      | ST          |                                       |
| RPC9          | 55              | 41               | I/O      | ST          |                                       |
| RPC10         | 45              | 33               | I/O      | ST          |                                       |
| RPC12         | 39              | 27               | I/O      | ST          |                                       |
| RPC13         | 47              | 35               | I/O      | ST          |                                       |
| RPC15         | 40              | 28               | I/O      | ST          |                                       |
| <b>PORT D</b> |                 |                  |          |             |                                       |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note 1:** Only Available on "MC" variants.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-1: PORTA THROUGH PORTG REMAPPABLE PERIPHERAL DESCRIPTIONS (CONTINUED) (CONTINUED)**

| Pin Name      | Pin Number      |                  | Pin Type | Buffer Type | Description                           |
|---------------|-----------------|------------------|----------|-------------|---------------------------------------|
|               | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                       |
| RPD5          | 53              | —                | I/O      | ST          | Remappable PORTD Peripheral Functions |
| RPD6          | 54              | —                | I/O      | ST          |                                       |
| <b>PORT E</b> |                 |                  |          |             |                                       |
| RPE0          | 34              | —                | I/O      | ST          | Remappable PORTE Peripheral Functions |
| RPE1          | 35              | —                | I/O      | ST          |                                       |
| RPE14         | 29              | —                | I/O      | ST          |                                       |
| RPE15         | 30              | —                | I/O      | ST          |                                       |
| <b>PORT F</b> |                 |                  |          |             |                                       |
| RPF0          | 58              | —                | I/O      | ST          | Remappable PORTF Peripheral Functions |
| RPF1          | 59              | —                | I/O      | ST          |                                       |
| <b>PORT G</b> |                 |                  |          |             |                                       |
| RPG6          | 4               | —                | I/O      | ST          | Remappable PORTG Peripheral Functions |
| RPG7          | 5               | —                | I/O      | ST          |                                       |
| RPG8          | 6               | —                | I/O      | ST          |                                       |
| RPG9          | 8               | —                | I/O      | ST          |                                       |

**Legend:** CMOS = CMOS-compatible input or output  
 ST = Schmitt Trigger input with CMOS levels  
 TTL = Transistor-transistor Logic input buffer

Analog = Analog input  
 O = Output  
 PPS = Peripheral Pin Select

P = Power  
 I = Input

**Note 1:** Only Available on “MC” variants.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-2: PORTA THROUGH PORT G PINOUT I/O DESCRIPTIONS**

| Pin Name      | Pin Number      |                  | Pin Type | Buffer Type | Description                       |
|---------------|-----------------|------------------|----------|-------------|-----------------------------------|
|               | 64-pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                   |
| <b>PORT A</b> |                 |                  |          |             |                                   |
| RA0           | 13              | 9                | I/O      | ST          | PORTA is a bidirectional I/O port |
| RA1           | 14              | 10               | I/O      | ST          |                                   |
| RA4           | 33              | 25               | I/O      | ST          |                                   |
| RA7           | 1               | 1                | I/O      | ST          |                                   |
| RA8           | 31              | 23               | I/O      | ST          |                                   |
| RA10          | 64              | 48               | I/O      | ST          |                                   |
| RA11          | 12              | 8                | I/O      | ST          |                                   |
| RA12          | 11              | 7                | I/O      | ST          |                                   |
| RA14          | 36              | ---              | I/O      | ST          |                                   |
| RA15          | 37              | ---              | I/O      | ST          |                                   |
| <b>PORT B</b> |                 |                  |          |             |                                   |
| RB0           | 15              | 11               | I/O      | ST          | PORTB is a bidirectional I/O port |
| RB1           | 16              | 12               | I/O      | ST          |                                   |
| RB2           | 17              | 13               | I/O      | ST          |                                   |
| RB3           | 18              | 14               | I/O      | ST          |                                   |
| RB4           | 32              | 24               | I/O      | ST          |                                   |
| RB5           | 43              | 31               | I/O      | ST          |                                   |
| RB6           | 44              | 32               | I/O      | ST          |                                   |
| RB7           | 46              | 34               | I/O      | ST          |                                   |
| RB8           | 48              | 36               | I/O      | ST          |                                   |
| RB9           | 49              | 37               | I/O      | ST          |                                   |
| RB10          | 60              | 44               | I/O      | ST          |                                   |
| RB11          | 61              | 45               | I/O      | ST          |                                   |
| RB12          | 62              | 46               | I/O      | ST          |                                   |
| RB13          | 63              | 47               | I/O      | ST          |                                   |
| RB14          | 2               | 2                | I/O      | ST          |                                   |
| RB15          | 3               | 3                | I/O      | ST          |                                   |
| <b>PORT C</b> |                 |                  |          |             |                                   |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-2: PORTA THROUGH PORT G PINOUT I/O DESCRIPTIONS (CONTINUED)**

|  |    |     |     |    |                                   |
|--|----|-----|-----|----|-----------------------------------|
| RC0  | 21 | 17  | I/O | ST | PORTC is a bidirectional I/O port |
| RC1  | 22 | 18  | I/O | ST |                                   |
| RC2  | 23 | 19  | I/O | ST |                                   |
| RC6  | 50 | 38  | I/O | ST |                                   |
| RC7  | 51 | 39  | I/O | ST |                                   |
| RC8  | 52 | 40  | I/O | ST |                                   |
| RC9  | 55 | 41  | I/O | ST |                                   |
| RC10   | 45 | 33  | I/O | ST |                                   |
| RC11   | 24 | 20  | I/O | ST |                                   |
| RC12   | 39 | 27  | I/O | ST |                                   |
| RC13   | 47 | 35  | I/O | ST |                                   |
| RC15   | 40 | 28  | I/O | ST |                                   |
| <b>PORT D</b>  |    |     |     |    |                                   |
| RD5  | 53 | --- | I/O | ST | PORTD is a bidirectional I/O port |
| RD6  | 54 | --- | I/O | ST |                                   |
| RD8  | 42 | 30  | I/O | ST |                                   |
| <b>PORT E</b>  |    |     |     |    |                                   |
| RE0  | 34 | --- | I/O | ST | PORTE is a bidirectional I/O port |
| RE1  | 35 | --- | I/O | ST |                                   |
| RE12   | 27 | --- | I/O | ST |                                   |
| RE13   | 28 | --- | I/O | ST |                                   |
| RE14   | 29 | --- | I/O | ST |                                   |
| RE15   | 30 | --- | I/O | ST |                                   |
| <b>PORT F</b>  |    |     |     |    |                                   |
| RF0  | 58 | --- | I/O | ST | PORTF is a bidirectional I/O port |
| RF1  | 59 | --- | I/O | ST |                                   |
| <b>PORT G</b>  |    |     |     |    |                                   |
| RG6  | 4  | --- | I/O | ST | PORTG is a bidirectional I/O port |
| RG7  | 5  | --- | I/O | ST |                                   |
| RG8  | 6  | --- | I/O | ST |                                   |
| RG9  | 8  | --- | I/O | ST |                                   |
| <p><b>Legend:</b> CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power<br/> ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input<br/> TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select</p> <p><b>Note 1:</b> Only Available on "MC" variants.</p> |    |     |     |    |                                   |



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-3: OSCILLATOR AND CLOCK PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description  |
|----------|-----------------|------------------|----------|-------------|--|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |  |
| CLKI     | 39              | 27               | I        | ST          | External clock source input. Always associated with OSC1 pin function.   |
| CLKO     | 40              | 28               | O        | CMOS        | Oscillator crystal output. Connects to crystal in Crystal Oscillator mode. Optionally functions as CLKO in FRC and EC modes. Always associated with OSC2 pin function. |
| OSCI     | 39              | 27               | I        | ST/CMOS     | Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise   |
| OSCO     | 40              | 28               | O        | CMOS        | Oscillator crystal output. Connects to crystal in Crystal Oscillator mode. Optionally functions as CLKO in FRC and EC modes.   |
| SOSCI    | 47              | 35               | I        | CMOS        | 32.768 kHz low-power crystal input   |
| SOSCO    | 48              | 36               | O/I      | ST/CMOS     | 32.768 low-power crystal output or 32.768 clock oscillator input when SOSSEN is disabled.  |
| REFCLKI  | PPS             | PPS              | I        | CMOS        | One of several alternate REFCLKOx user-selectable input clock sources.<br><br>Reference Clock Generator Outputs 1-4  |
| REFCLKO1 | PPS             | PPS              | O        | CMOS        |  |
| REFCLKO2 | PPS             | PPS              | O        | CMOS        |  |
| REFCLKO3 | PPS             | PPS              | O        | CMOS        |  |
| REFCLKO4 | PPS             | PPS              | O        | CMOS        |  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note 1:** Only Available on "MC" variants.

**TABLE 1-4: INPUT CAPTURE PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description                |
|----------|-----------------|------------------|----------|-------------|----------------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                            |
| IC1      | PPS             | PPS              | I        | ST          | Input Capture Inputs 1 - 9 |
| IC2      | PPS             | PPS              | I        | ST          |                            |
| IC3      | PPS             | PPS              | I        | ST          |                            |
| IC4      | PPS             | PPS              | I        | ST          |                            |
| IC5      | PPS             | PPS              | I        | ST          |                            |
| IC6      | PPS             | PPS              | I        | ST          |                            |
| IC7      | PPS             | PPS              | I        | ST          |                            |
| IC8      | PPS             | PPS              | I        | ST          |                            |
| IC9      | PPS             | PPS              | I        | ST          |                            |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-5: OUTPUT COMPARE PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description                  |
|----------|-----------------|------------------|----------|-------------|------------------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                              |
| OC1      | PPS             | PPS              | O        | CMOS        | Output Compare 1 - 9         |
| OC2      | PPS             | PPS              | O        | CMOS        |                              |
| OC3      | PPS             | PPS              | O        | CMOS        |                              |
| OC4      | PPS             | PPS              | O        | CMOS        |                              |
| OC5      | PPS             | PPS              | O        | CMOS        |                              |
| OC6      | PPS             | PPS              | O        | CMOS        |                              |
| OC7      | PPS             | PPS              | O        | CMOS        |                              |
| OC8      | PPS             | PPS              | O        | CMOS        |                              |
| OC9      | PPS             | PPS              | O        | CMOS        |                              |
| OCFA     | PPS             | PPS              | I        | ST          | Output Compare Fault A Input |
| OCFB     | PPS             | PPS              | I        | ST          | Output Compare Fault B Input |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**TABLE 1-6: EXTERNAL INTERRUPTS PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description          |
|----------|-----------------|------------------|----------|-------------|----------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                      |
| INT0     | 46              | 34               | I        | ST          | External Interrupt 0 |
| INT1     | PPS             | PPS              | I        | ST          | External Interrupt 1 |
| INT2     | PPS             | PPS              | I        | ST          | External Interrupt 2 |
| INT3     | PPS             | PPS              | I        | ST          | External Interrupt 3 |
| INT4     | PPS             | PPS              | I        | ST          | External Interrupt 4 |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-7: UART1 THROUGH UART2 PINOUT I/O DESCRIPTIONS**

| Pin Name                                      | Pin Number      |                  | Pin Type | Buffer Type | Description           |
|---|-----------------|------------------|----------|-------------|-----------------------|
|   | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                       |
| Universal Asynchronous Receiver Transmitter 1 |                 |                  |          |             |                       |
| U1RX  | PPS             | PPS              | I        | St          | UART1 Receive         |
| U1TX  | PPS             | PPS              | O        | CMOS        | UART1 Transmit        |
| U1CTS   | PPS             | PPS              | I        | ST          | UART1 Clear to Send   |
| U1RTS   | PPS             | PPS              | O        | CMOS        | UART1 Request to Send |
| Universal Asynchronous Receiver Transmitter 2 |                 |                  |          |             |                       |
| U2RX  | PPS             | PPS              | I        | ST          | UART2 Receive         |
| U2TX  | PPS             | PPS              | O        | CMOS        | UART2 Transmit        |
| U2CTS   | PPS             | PPS              | I        | ST          | UART2 Clear to Send   |
| U2RTS   | PPS             | PPS              | O        | CMOS        | UART2 Request to Send |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**TABLE 1-8: SPI1 THROUGH SPI2 PINOUT I/O DESCRIPTIONS**

| Pin Name                      | Pin Number      |                  | Pin Type | Buffer Type | Description                                   |
|-------------------------------|-----------------|------------------|----------|-------------|---|
|                               | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |   |
| Serial Peripheral Interface 1 |                 |                  |          |             |   |
| SCK1                          | 46              | 34               | I/O      | ST/CMOS     | SPI1 Synchronous Serial Clock Input/Output    |
| SDI1                          | PPS             | PPS              | I        | ST          | SPI1 Data In                                  |
| SDO1                          | PPS             | PPS              | O        | CMOS        | SPI1 Data Out                                 |
| SS1                           | PPS             | PPS              | I/O      | ST/CMOS     | SPI1 Slave Synchronization Or Frame Pulse I/O |
| Serial Peripheral Interface 2 |                 |                  |          |             |   |
| SCK2                          | PPS             | PPS              | I/O      | ST/CMOS     | SPI2 Synchronous Serial Clock Input/Output    |
| SDI2                          | PPS             | PPS              | I        | ST          | SPI2 Data In                                  |
| SDO2                          | PPS             | PPS              | O        | CMOS        | SPI2 Data Out                                 |
| SS2                           | PPS             | PPS              | I/O      | ST/CMOS     | SPI2 Slave Synchronization Or Frame Pulse I/O |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**TABLE 1-9: I2C1 THROUGH I2C2 PINOUT I/O DESCRIPTIONS**

| Pin Name                   | Pin Number      |                  | Pin Type | Buffer Type | Description                                |
|----------------------------|-----------------|------------------|----------|-------------|--|
|                            | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |  |
| Inter-Integrated Circuit 1 |                 |                  |          |             |  |
| SCL1                       | 44              | 32               | I/O      | ST          | I2C1 Synchronous Serial Clock Input/Output |
| SDA1                       | 43              | 31               | I/O      | ST          | I2C1 Synchronous Serial Data Input/Output  |
| Inter-Integrated Circuit 2 |                 |                  |          |             |  |
| SCL2                       | 32              | 24               | I/O      | ST          | I2C2 Synchronous Serial Clock Input/Output |
| SDA2                       | 31              | 23               | I/O      | ST          | I2C2 Synchronous Serial Data Input/Output  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-10: TIMER1 THROUGH TIMER9 AND RTCC PINOUT I/O DESCRIPTIONS**

| Pin Name              | Pin Number      |                  | Pin Type | Buffer Type | Description                          |
|-----------------------|-----------------|------------------|----------|-------------|--------------------------------------|
|                       | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                      |
| Timer1 through Timer9 |                 |                  |          |             |                                      |
| T1CK                  | 48              | 36               | I        | ST          | Timer1 External Clock Input          |
| T2CK                  | PPS             | PPS              | I        | ST          | Timer2 External Clock Input          |
| T3CK                  | PPS             | PPS              | I        | ST          | Timer3 External Clock Input          |
| T4CK                  | PPS             | PPS              | I        | ST          | Timer4 External Clock Input          |
| T5CK                  | PPS             | PPS              | I        | ST          | Timer5 External Clock Input          |
| T6CK                  | PPS             | PPS              | I        | ST          | Timer6 External Clock Input          |
| T7CK                  | PPS             | PPS              | I        | ST          | Timer7 External Clock Input          |
| T8CK                  | PPS             | PPS              | I        | ST          | Timer8 External Clock Input          |
| T9CK                  | PPS             | PPS              | I        | ST          | Timer9 External Clock Input          |
| RTCC                  | 18              | 14               | O        | CMOS        | Real-Time Clock Alarm/Seconds Output |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-11: COMPARATOR 1 THROUGH COMPARATOR 5 PINOUT I/O DESCRIPTIONS**

| Pin Name     | Pin Number      |                  | Pin Type | Buffer Type | Description                    |
|--------------|-----------------|------------------|----------|-------------|--------------------------------|
|              | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                |
| Comparator 1 |                 |                  |          |             |                                |
| C1IN1-       | 18              | 14               | I        | Analog      | Comparator #1 Negative Input 1 |
| C1IN1+       | 17              | 13               | I        | Analog      | Comparator #1 Positive Input 1 |
| C1IN2-       | 24              | 20               | I        | Analog      | Comparator #1 Negative Input 2 |
| C1IN3-       | 17              | 13               | I        | Analog      | Comparator #1 Negative Input 3 |
| C1IN4-       | 16              | 12               | I        | Analog      | Comparator #1 Negative Input 4 |
| Comparator 2 |                 |                  |          |             |                                |
| C2IN1-       | 15              | 11               | I        | Analog      | Comparator #2 Negative Input 1 |
| C2IN1+       | 14              | 10               | I        | Analog      | Comparator #2 Positive Input 1 |
| C2IN2-       | 27              | -                | I        | Analog      | Comparator #2 Negative Input 2 |
| C2IN3-       | 17              | 13               | I        | Analog      | Comparator #2 Negative Input 3 |
| C2IN4-       | 13              | 9                | I        | Analog      | Comparator #2 Negative Input 4 |
| Comparator 3 |                 |                  |          |             |                                |
| C3IN1-       | 22              | 18               | I        | Analog      | Comparator #3 Negative Input 1 |
| C3IN1+       | 23              | 19               | I        | Analog      | Comparator #3 Positive Input 1 |
| C3IN2-       | 28              | -                | I        | Analog      | Comparator #3 Negative Input 2 |
| C3IN3-       | 23              | 19               | I        | Analog      | Comparator #3 Negative Input 3 |
| C3IN4-       | 21              | 17               | I        | Analog      | Comparator #3 Negative Input 4 |
| Comparator 4 |                 |                  |          |             |                                |
| C4IN1-       | 22              | 18               | I        | Analog      | Comparator #4 Negative Input 1 |
| C4IN1+       | 21              | 17               | I        | Analog      | Comparator #4 Positive Input 1 |
| C4IN2-       | 16              | 12               | I        | Analog      | Comparator #4 Negative Input 2 |
| C4IN3-       | 13              | 9                | I        | Analog      | Comparator #4 Negative Input 3 |
| C4IN4-       | 21              | 17               | I        | Analog      | Comparator #4 Negative Input 4 |
| Comparator 5 |                 |                  |          |             |                                |
| C5IN1-       | 49              | 37               | I        | Analog      | Comparator #5 Negative Input 1 |
| C5IN1+       | 33              | 25               | I        | Analog      | Comparator #5 Positive Input 1 |
| C5IN2-       | 27              | -                | I        | Analog      | Comparator #5 Negative Input 2 |
| C5IN3-       | 33              | 25               | I        | Analog      | Comparator #5 Negative Input 3 |
| C5IN4-       | 46              | 34               | I        | Analog      | Comparator #5 Negative Input 4 |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-12: OP AMP 1 THROUGH OP-AMP 5 PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description     |
|----------|-----------------|------------------|----------|-------------|-----------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                 |
| Op amp 1 |                 |                  |          |             |                 |
| OA1IN-   | 18              | 14               | I        | Analog      | Op amp 1 Input  |
| OA1IN+   | 17              | 13               | I        | Analog      | Op amp 1 Input  |
| OA1OUT   | 16              | 12               | O        | Analog      | Op amp 1 Output |
| Op amp 2 |                 |                  |          |             |                 |
| OA2IN-   | 15              | 11               | I        | Analog      | Op amp 2 Input  |
| OA2IN+   | 14              | 10               | I        | Analog      | Op amp 2 Input  |
| OA2OUT   | 13              | 9                | O        | Analog      | Op amp 2 Output |
| Op amp 3 |                 |                  |          |             |                 |
| OA3IN-   | 22              | 18               | I        | Analog      | Op amp 3 Input  |
| OA3IN+   | 23              | 19               | I        | Analog      | Op amp 3 Input  |
| OA3OUT   | 21              | 17               | O        | Analog      | Op amp 3 Output |
| Op amp 5 |                 |                  |          |             |                 |
| OA5IN-   | 49              | 37               | I        | Analog      | Op amp 5 Input  |
| OA5IN+   | 33              | 25               | I        | Analog      | Op amp 5 Input  |
| OA5OUT   | 46              | 34               | O        | Analog      | Op amp 5 Output |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**TABLE 1-13: CAN\_FD PINOUT I/O DESCRIPTIONS**

| Pin Name            | Pin Number      |                  | Pin Type | Buffer Type | Description             |
|---------------------|-----------------|------------------|----------|-------------|-------------------------|
|                     | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                         |
| C1TX <sup>(1)</sup> | PPS             | PPS              | O        | CMOS        | CANFD1 Bus Transmit Pin |
| C1RX <sup>(1)</sup> | PPS             | PPS              | I        | ST          | CANFD1 Bus Receive Pin  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note 1:** Not available on “GPG” variants.

**TABLE 1-14: CTMU PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description  |
|----------|-----------------|------------------|----------|-------------|--|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |  |
| CTCMP    | 18              | 14               | I        | Analog      | CTMU external capacitor input for pulse generation |
| CTED1    | 16              | 12               | I        | ST          | CTMU External Edge/Level Input 1                   |
| CTED2    | 15              | 11               | I        | ST          | CTMU External Edge/Level Input 2                   |
| CTPLS    | 63              | 47               | O        | CMOS        | CTMU Pulse Generator Output                        |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-15: DAC1 AND DAC2 PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description               |
|----------|-----------------|------------------|----------|-------------|---------------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                           |
| DAC1     | 45              | 33               | O        | Analog      | Control 12bit DAC1 Output |
| DAC2     | 31              | 23               | O        | Analog      | Control 12bit DAC2 Output |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**TABLE 1-16: MCPWM1 THROUGH MCPWM9 PINOUT I/O DESCRIPTIONS (MOTOR CONTROL DEVICES ONLY)**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description             |
|----------|-----------------|------------------|----------|-------------|-------------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                         |
| PWM1H    | 2               | 2                | O        | CMOS        | MCPWM1 High Side Output |
| PWM1L    | 3               | 3                | O        | CMOS        | MCPWM1 Low Side Output  |
| PWM2H    | 62              | 46               | O        | CMOS        | MCPWM2 High Side Output |
| PWM2L    | 63              | 47               | O        | CMOS        | MCPWM2 Low Side Output  |
| PWM3H    | 60              | 44               | O        | CMOS        | MCPWM3 High Side Output |
| PWM3L    | 61              | 45               | O        | CMOS        | MCPWM3 Low Side Output  |
| PWM4H    | 64              | 48               | O        | CMOS        | MCPWM4 High Side Output |
| PWM4L    | 1               | 1                | O        | CMOS        | MCPWM4 Low Side Output  |
| PWM5H    | 52              | 40               | O        | CMOS        | MCPWM5 High Side Output |
| PWM5L    | 55              | 41               | O        | CMOS        | MCPWM5 Low Side Output  |
| PWM6H    | 50              | 38               | O        | CMOS        | MCPWM6 High Side Output |
| PWM6L    | 51              | 39               | O        | CMOS        | MCPWM6 Low Side Output  |
| PWM7H    | 5               | -                | O        | CMOS        | MCPWM7 High Side Output |
| PWM7L    | 4               | -                | O        | CMOS        | MCPWM7 Low Side Output  |
| PWM8H    | 58              | -                | O        | CMOS        | MCPWM8 High Side Output |
| PWM8L    | 59              | -                | O        | CMOS        | MCPWM8 Low Side Output  |
| PWM9H    | 53              | -                | O        | CMOS        | MCPWM9 High Side Output |
| PWM9L    | 54              | -                | O        | CMOS        | MCPWM9 Low Side Output  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note:** PWM features are only supported on “MC” variants.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-17: MCPWM FAULT, CURRENT-LIMIT, AND DEAD TIME COMPENSATION PINOUT I/O DESCRIPTIONS (MOTOR CONTROL DEVICES ONLY)**

| Pin Name | Pin Number       |                   | Pin Type | Buffer Type | Description                              |
|----------|------------------|-------------------|----------|-------------|--|
|          | 64-Pin TQFP/ QFN | 48-Pin TQFP/ VQFN |          |             |  |
| FLT3     | 23               | 19                | I        | ST          | PWM Fault / Dead Time Comp Input Control |
| FLT4     | 24               | 20                | I        | ST          |  |
| FLT5     | 27               | -                 | I        | ST          |  |
| FLT6     | 28               | -                 | I        | ST          |  |
| FLT7     | 29               | -                 | I        | ST          |  |
| FLT8     | 30               | -                 | I        | ST          |  |
| FLT12    | 8                | -                 | I        | ST          |  |
| FLT13    | 11               | 7                 | I        | ST          |  |
| FLT14    | 12               | 8                 | I        | ST          |  |
| FLT15    | 32               | 24                | I        | ST          |  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note:** MCPWM fault features are only supported on “MC” variants.

**TABLE 1-18: QE1 THROUGH QE3 PINOUT I/O DESCRIPTIONS (MOTOR CONTROL DEVICES ONLY)**

| Pin Name                       | Pin Number       |                   | Pin Type | Buffer Type | Description  |
|--------------------------------|------------------|-------------------|----------|-------------|--|
|                                | 64-Pin TQFP/ QFN | 48-Pin TQFP/ VQFN |          |             |  |
| Quadrature Encoder Interface 1 |                  |                   |          |             |  |
| QEA1                           | PPS              | PPS               | I        | ST          | QE1 Phase A Input in QE1 mode                        |
| QEB1                           | PPS              | PPS               | I        | ST          | QE1 Phase B Input in QE1 Mode. Auxiliary Timer mode. |
| INDX1                          | PPS              | PPS               | I        | ST          | QE1 Index Pulse Input                                |
| HOME1                          | PPS              | PPS               | I        | ST          | QE1 Position Counter Input Capture Trigger           |
| QEICMP1                        | PPS              | PPS               | I        | ST          | QE1 Capture Compare Match Output                     |
| Quadrature Encoder Interface 2 |                  |                   |          |             |  |
| QEA2                           | PPS              | PPS               | I        | ST          | QE2 Phase A Input in QE1 mode                        |
| QEB2                           | PPS              | PPS               | I        | ST          | QE2 Phase B Input in QE1 Mode. Auxiliary Timer mode. |
| INDX2                          | PPS              | PPS               | I        | ST          | QE2 Index Pulse Input                                |
| HOME2                          | PPS              | PPS               | I        | ST          | QE2 Position Counter Input Capture Trigger           |
| QEICMP2                        | PPS              | PPS               | I        | ST          | QE2 Capture Compare Match Output                     |
| Quadrature Encoder Interface 3 |                  |                   |          |             |  |
| QEA3                           | PPS              | PPS               | I        | ST          | QE3 Phase A Input in QE1 mode                        |
| QEB3                           | PPS              | PPS               | I        | ST          | QE3 Phase B Input in QE1 Mode. Auxiliary Timer mode. |
| INDX3                          | PPS              | PPS               | I        | ST          | QE3 Index Pulse Input                                |
| HOME3                          | PPS              | PPS               | I        | ST          | QE3 Position Counter Input Capture Trigger           |
| QEICMP3                        | PPS              | PPS               | I        | ST          | QE3 Capture Compare Match Output                     |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note:** QE1 features are only supported on “MC” variants.



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-19: POWER, GROUND, HLVD AND VOLTAGE REFERENCE PINOUT I/O DESCRIPTIONS**

| Pin Name                  | Pin Number                         |                               | Pin Type | Buffer Type | Description  |
|---------------------------|------------------------------------|-------------------------------|----------|-------------|--|
|                           | 64-Pin TQFP/QFN                    | 48-Pin TQFP/VQFN              |          |             |  |
| AVDD                      | 19                                 | 15                            | P        | P           | Positive supply for analog modules. This pin must be connected at all times.               |
| AVSS                      | 20                                 | 16                            | P        | P           | Ground reference for analog modules. This pin must be connected at all times               |
| VDD                       | 10, 26, 38, 42 <sup>(1)</sup> , 57 | 6,22,26,30 <sup>(1)</sup> ,43 | P        | P           | Positive supply for peripheral logic and I/O pins. This pin must be connected at all times |
| VSS                       | 9, 25, 41, 56                      | 5, 21, 29, 42                 | P        | P           | Ground reference for logic, and I/O pins. This pin must be connected at all times          |
| Voltage References        |                                    |                               |          |             |  |
| VREF+                     | 16                                 | 12                            | I        | Analog      | Analog Voltage Reference (High) Input  |
| VREF-                     | 15                                 | 11                            | I        | Analog      | Analog Voltage Reference (Low) Input   |
| High / Low Voltage Detect |                                    |                               |          |             |  |
| LVDIN                     | 49                                 | 37                            | I        | Analog      | High / Low Voltage detect input  |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

**Note 1:** Only available on “GP” variants.

**TABLE 1-20: JTAG, TRACE, MASTER CLEAR AND PROGRAMMING/DEBUGGING PINOUT I/O DESCRIPTIONS**

| Pin Name              | Pin Number      |                  | Pin Type | Buffer Type | Description   |
|-----------------------|-----------------|------------------|----------|-------------|---|
|                       | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |   |
| JTAG Interface        |                 |                  |          |             |   |
| TCK                   | 1               | 1                | I        | ST          | JTAG Test Clock Input Pin   |
| TDI                   | 31              | 23               | I        | ST          | JTAG Test Data Input Pin  |
| TDO                   | 64              | 48               | O        | CMOS        | JTAG Test Data Output Pin   |
| TMS                   | 49              | 37               | I        | ST          | JTAG Test Mode Select Pin   |
| Trace Interface       |                 |                  |          |             |   |
| TRCLK                 | 50              | -                | O        | CMOS        | Trace Clock   |
| TRD0                  | 51              | -                | O        | CMOS        | Trace Data bits 0-3<br>Trace support is available through the MPLAB® REAL ICE™ In-circuit Emulator. |
| TRD1                  | 52              | -                | O        | CMOS        |   |
| TRD2                  | 53              | -                | O        | CMOS        |   |
| TRD3                  | 54              | -                | O        | CMOS        |   |
| Programming/Debugging |                 |                  |          |             |   |
| PGED1                 | 18              | 14               | I/O      | ST          | Data I/O pin for Programming/Debugging Communication Channel 1                                      |
| PGEC1                 | 17              | 13               | I        | ST          | Clock input pin for Programming/Debugging Communication Channel 1                                   |
| PGED2                 | 43              | 31               | I/O      | ST          | Data I/O pin for Programming/Debugging Communication Channel 2                                      |
| PGEC2                 | 44              | 32               | I        | ST          | Clock input pin for Programming/Debugging Communication Channel 2                                   |
| PGED3                 | 15              | 11               | I/O      | ST          | Data I/O pin for Programming/Debugging Communication Channel 3                                      |
| PGEC3                 | 16              | 12               | I        | ST          | Clock input pin for Programming/Debugging Communication Channel 3                                   |
| Master Clear          |                 |                  |          |             |   |
| MCLR                  | 7               | 4                | I        | ST          | Master Clear (Reset) input. This pin is an active-low Reset to the device.                          |

**Legend:** CMOS = CMOS-compatible input or output      Analog = Analog input      P = Power  
 ST = Schmitt Trigger input with CMOS levels      O = Output      I = Input  
 TTL = Transistor-transistor Logic input buffer      PPS = Peripheral Pin Select

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-21: CONFIGURABLE LOGIC CELL PINOUT I/O DESCRIPTIONS**

| Pin Name                    | Pin Number      |                  | Pin Type | Buffer Type | Description                             |
|-----------------------------|-----------------|------------------|----------|-------------|---|
|                             | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |   |
| Configurable Logic Module 1 |                 |                  |          |             | Configurable Logic Inputs & Outputs 1-4 |
| CLCINA                      | PPS             | PPS              | I        | ST          |   |
| CLC01                       | PPS             | PPS              | O        | CMOS        |   |
| Configurable Logic Module 2 |                 |                  |          |             |   |
| CLCINB                      | PPS             | PPS              | I        | ST          |   |
| CLC02                       | PPS             | PPS              | O        | CMOS        |   |
| Configurable Logic Module 3 |                 |                  |          |             |   |
| CLCINC                      | PPS             | PPS              | I        | ST          |   |
| CLC03                       | PPS             | PPS              | O        | CMOS        |   |
| Configurable Logic Module 4 |                 |                  |          |             |   |
| CLCIND                      | PPS             | PPS              | I        | ST          |   |
| CLC04                       | PPS             | PPS              | O        | CMOS        |   |

**Legend:** CMOS = CMOS-compatible input or output  
 ST = Schmitt Trigger input with CMOS levels  
 TTL = Transistor-transistor Logic input buffer  
 Analog = Analog input  
 O = Output  
 PPS = Peripheral Pin Select  
 P = Power  
 I = Input

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-22: CAPACITIVE VOLTAGE DIVIDER PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description                       |
|----------|-----------------|------------------|----------|-------------|-----------------------------------|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |                                   |
| CVD6     | 21              | 17               | I        | Analog      | Capacitive Voltage Divider Inputs |
| CVD7     | 22              | 18               | I        | Analog      |                                   |
| CVD8     | 23              | 19               | I        | Analog      |                                   |
| CVD9     | 12              | 8                | I        | Analog      |                                   |
| CVD10    | 11              | 7                | I        | Analog      |                                   |
| CVD11    | 24              | 20               | I        | Analog      |                                   |
| CVD12    | 27              | -                | I        | Analog      |                                   |
| CVD13    | 28              | -                | I        | Analog      |                                   |
| CVD14    | 29              | -                | I        | Analog      |                                   |
| CVD15    | 30              | -                | I        | Analog      |                                   |
| CVD16    | 8               | -                | I        | Analog      |                                   |
| CVD17    | 6               | -                | I        | Analog      |                                   |
| CVD18    | 5               | -                | I        | Analog      |                                   |
| CVD19    | 4               | -                | I        | Analog      |                                   |
| CVD24    | 33              | 25               | I        | Analog      |                                   |
| CVD25    | 46              | 34               | I        | Analog      |                                   |
| CVD26    | 31              | 23               | I        | Analog      |                                   |
| CVD27    | 49              | 37               | I        | Analog      |                                   |
| CVD40    | 34              | -                | I        | Analog      |                                   |
| CVD41    | 35              | -                | I        | Analog      |                                   |
| CVD46    | 36              | -                | I        | Analog      |                                   |
| CVD47    | 37              | -                | I        | Analog      |                                   |
| CVD48    | 45              | 33               | I        | Analog      |                                   |
| CVD49    | 39              | 27               | I        | Analog      |                                   |

**Legend:** CMOS = CMOS-compatible input or output  
 ST = Schmitt Trigger input with CMOS levels  
 TTL = Transistor-transistor Logic input buffer

Analog = Analog input  
 O = Output  
 PPS = Peripheral Pin Select

P = Power  
 I = Input

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 1-23: ADC0 - ADC5, ADC7 PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number      |                  | Pin Type | Buffer Type | Description  |
|----------|-----------------|------------------|----------|-------------|--|
|          | 64-Pin TQFP/QFN | 48-Pin TQFP/VQFN |          |             |  |
| AN0      | 13              | 9                | I        | Analog      | Dedicated Class_1 ADC0 module Analog Input Channel |
| AN1      | 14              | 10               | I        | Analog      | Dedicated Class_1 ADC1 module Analog Input Channel |
| AN2      | 15              | 11               | I        | Analog      | Dedicated Class_1 ADC2 module Analog Input Channel |
| AN3      | 16              | 12               | I        | Analog      | Dedicated Class_1 ADC3 module Analog Input Channel |
| AN4      | 17              | 13               | I        | Analog      | Dedicated Class_1 ADC4 module Analog Input Channel |
| AN5      | 18              | 14               | I        | Analog      | Dedicated Class_1 ADC5 module Analog Input Channel |
| AN6      | 21              | 17               | I        | Analog      | Shared ADC7 module Analog Input Channels           |
| AN7      | 22              | 18               | I        | Analog      |  |
| AN8      | 23              | 19               | I        | Analog      |  |
| AN9      | 12              | 8                | I        | Analog      |  |
| AN10     | 11              | 7                | I        | Analog      |  |
| AN11     | 24              | 20               | I        | Analog      |  |
| AN12     | 27              | -                | I        | Analog      |  |
| AN13     | 28              | -                | I        | Analog      |  |
| AN14     | 29              | -                | I        | Analog      |  |
| AN15     | 30              | -                | I        | Analog      |  |
| AN16     | 8               | -                | I        | Analog      |  |
| AN17     | 6               | -                | I        | Analog      |  |
| AN18     | 5               | -                | I        | Analog      |  |
| AN19     | 4               | -                | I        | Analog      |  |
| AN24     | 33              | 25               | I        | Analog      |  |
| AN25     | 46              | 34               | I        | Analog      |  |
| AN26     | 31              | 23               | I        | Analog      |  |
| AN27     | 49              | 37               | I        | Analog      |  |
| AN40     | 34              | -                | I        | Analog      |  |
| AN41     | 35              | -                | I        | Analog      |  |
| AN46     | 36              | -                | I        | Analog      |  |
| AN47     | 37              | -                | I        | Analog      |  |
| AN48     | 45              | 33               | I        | Analog      |  |
| AN49     | 39              | 27               | I        | Analog      |  |

**Legend:** CMOS = CMOS-compatible input or output  
 ST = Schmitt Trigger input with CMOS levels  
 TTL = Transistor-transistor Logic input buffer

Analog = Analog input  
 O = Output  
 PPS = Peripheral Pin Select

P = Power  
 I = Input

# PIC32MK GPG/MCJ with CAN FD Family

## 2.0 GUIDELINES FOR GETTING STARTED WITH 32-BIT MCUS

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the documents listed in the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

### 2.1 Basic Connection Requirements

Getting started with the PIC32MK GPG/MCJ with CAN FD Family of 32-bit Microcontrollers (MCUs) requires attention to a minimal set of device pin connections before proceeding with development. The following is a list of pin names, which must always be connected:

- All VDD and VSS pins (see [2.2 “Decoupling Capacitors”](#))
- All AVDD and AVSS pins, even if the ADC module is not used (see [2.2 “Decoupling Capacitors”](#))
- MCLR pin (see [2.3 “Master Clear \(MCLR\) Pin”](#))
- PGECx/PGEDx pins, used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see [2.4 “ICSP Pins”](#))
- OSC1 and OSC2 pins, when external oscillator source is used (see [2.7 “External Oscillator Pins”](#))

The following pins may be required:

VREF+/VREF- pins, used when external voltage reference for the ADC module is implemented.

**Note:** The AVDD and AVSS pins must be connected, regardless of ADC use and the ADC voltage reference source.

### 2.2 Decoupling Capacitors

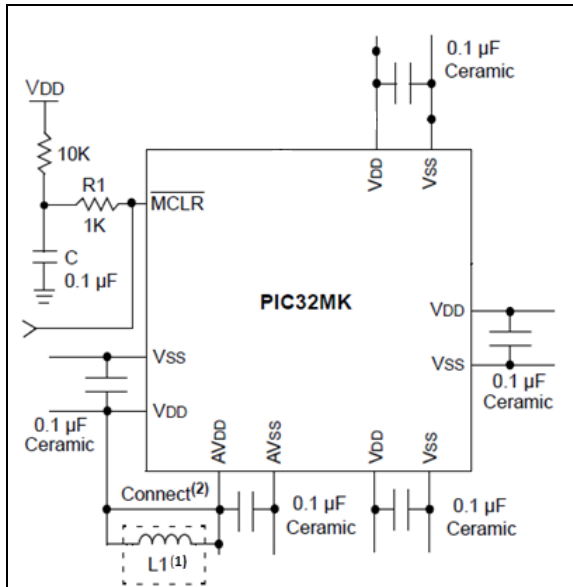
The use of decoupling capacitors on power supply pins, such as VDD, VSS, AVDD and AVSS is required, see [Figure 2-1](#).

Consider the following criteria when using decoupling capacitors:

- **Value and type of capacitor:** A value of 0.1  $\mu\text{F}$  (100 nF), 10-20V is recommended. The capacitor should be a low Equivalent Series Resistance (low-ESR) capacitor and have resonance frequency in the range of 20 MHz and higher. It is further recommended that ceramic capacitors be used.
- **Placement on the printed circuit board:** The decoupling capacitors should be placed as close to the pins as possible. It is recommended that the capacitors be placed on the same side of the board as the device. If space is constricted, the capacitor can be placed on another layer on the PCB using a via; however, ensure that the trace length from the pin to the capacitor is within one-quarter inch (6 mm) in length.
- **Handling high frequency noise:** If the board is experiencing high frequency noise, upward of tens of MHz, add a second ceramic-type capacitor in parallel to the above described decoupling capacitor. The value of the second capacitor can be in the range of 0.01  $\mu\text{F}$  to 0.001  $\mu\text{F}$ . Place this second capacitor next to the primary decoupling capacitor. In high-speed circuit designs, consider implementing a decade pair of capacitances as close to the power and ground pins as possible. For example, 0.1  $\mu\text{F}$  in parallel with 0.001  $\mu\text{F}$ .
- **Maximizing performance:** On the board layout from the power supply circuit, run the power and return traces to the decoupling capacitors first, and then to the device pins. This ensures that the decoupling capacitors are first in the power chain. Equally important is to keep the trace length between the capacitor and the power pins to a minimum thereby reducing PCB track inductance.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 2-1: RECOMMENDED MINIMUM CONNECTION**



**Note 1:** As an option, instead of a hard-wired connection, an inductor (L1) can be substituted between VDD and AVDD to improve ADC noise rejection. The inductor impedance must be less than 3Ω and the inductor capacity is greater than 10 mA.

Where:

$$f = \frac{FCNV}{2} \quad (\text{i.e., ADC conversion rate}/2)$$

$$f = \frac{1}{(2\pi\sqrt{LC})}$$

$$L = \left( \frac{1}{(2\pi f\sqrt{C})} \right)^2$$

## 2.2.1 BULK CAPACITORS

The use of a bulk capacitor is recommended to improve power supply stability. Typical values range from 4.7 μF to 47 μF. This capacitor should be located as close to the device as possible.

## 2.3 Master Clear ( $\overline{\text{MCLR}}$ ) Pin

The  $\overline{\text{MCLR}}$  pin provides two specific device functions:

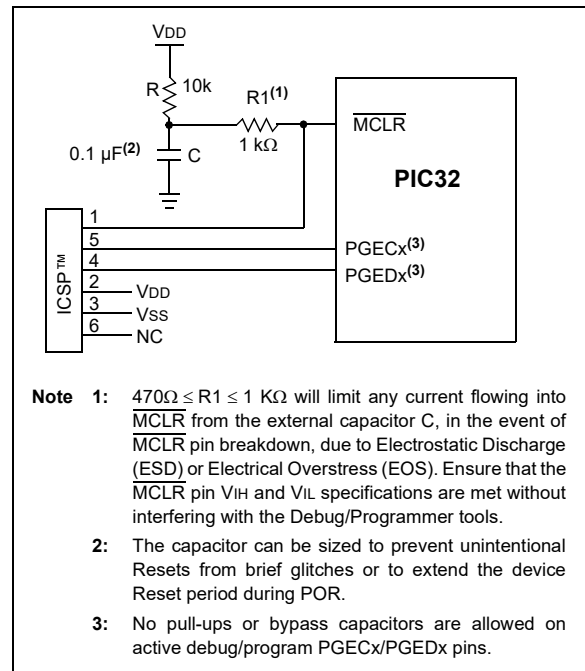
- Device Reset
- Device programming and debugging

Pulling The  $\overline{\text{MCLR}}$  pin low generates a device Reset. Figure 2-2 illustrates a typical MCLR circuit. During device programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the  $\overline{\text{MCLR}}$  pin. Consequently, specific voltage levels ( $V_{IH}$  and  $V_{IL}$ ) and fast signal transitions must not be adversely affected. Therefore, specific values of R and C will need to be adjusted based on the application and PCB requirements.

For example, as illustrated in Figure 2-2, it is recommended that the capacitor C, be isolated from the  $\overline{\text{MCLR}}$  pin during programming and debugging operations.

Place the components illustrated in Figure 2-2 within one-quarter inch (6 mm) from the  $\overline{\text{MCLR}}$  pin.

**FIGURE 2-2: EXAMPLE OF  $\overline{\text{MCLR}}$  PIN CONNECTIONS**



- Note 1:**  $470\Omega \leq R1 \leq 1\text{ K}\Omega$  will limit any current flowing into  $\overline{\text{MCLR}}$  from the external capacitor C, in the event of  $\overline{\text{MCLR}}$  pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the  $\overline{\text{MCLR}}$  pin  $V_{IH}$  and  $V_{IL}$  specifications are met without interfering with the Debug/Programmer tools.
- 2:** The capacitor can be sized to prevent unintentional Resets from brief glitches or to extend the device Reset period during POR.
- 3:** No pull-ups or bypass capacitors are allowed on active debug/program PGECx/PGEDx pins.

# PIC32MK GPG/MCJ with CAN FD Family

## 2.4 ICSP Pins

The PGECx and PGEDx pins are used for ICSP and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of Ohms, not to exceed 100 Ohms.

Pull-up resistors, series diodes and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin input voltage high ( $V_{IH}$ ) and input low ( $V_{IL}$ ) requirements.

Ensure that the “Communication Channel Select” (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to MPLAB® ICD 3 or MPLAB REAL ICE™.

For additional information on ICD 3 and REAL ICE connection requirements, refer to the following documents that are available on the Microchip web site.

- “Using MPLAB® ICD 3” (poster) DS50001765
- “MPLAB® ICD 3 Design Advisory” DS50001764
- “MPLAB® REAL ICE™ In-Circuit Debugger User’s Guide” DS50001616
- “Using MPLAB® REAL ICE™ Emulator” (poster) DS50001749

## 2.5 JTAG

The TMS, TDO, TDI and TCK pins are used for testing and debugging according to the Joint Test Action Group (JTAG) standard. It is recommended to keep the trace length between the JTAG connector and the JTAG pins on the device as short as possible. If the JTAG connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of Ohms, not to exceed 100 Ohms.

Pull-up resistors, series diodes and capacitors on the TMS, TDO, TDI and TCK pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin input voltage high ( $V_{IH}$ ) and input low ( $V_{IL}$ ) requirements.

## 2.6 Trace

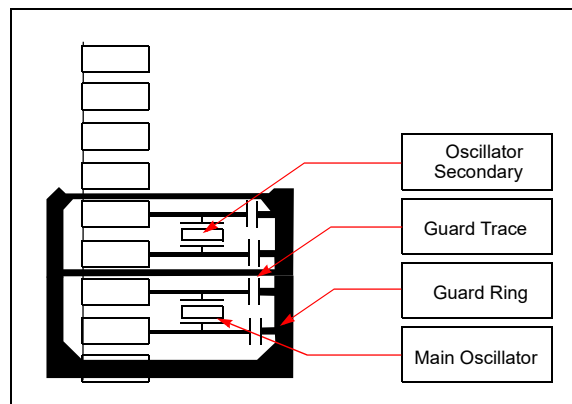
When present on select pin counts, the trace pins can be connected to a hardware trace-enabled programmer to provide a compressed real-time instruction trace. When used for trace, the TRD3, TRD2, TRD1, TRD0 and TRCLK pins should be dedicated for this use. The trace hardware requires a 22 Ohm series resistor between the trace pins and the trace connector.

## 2.7 External Oscillator Pins

Many MCUs have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to [Section 7.0 “Oscillator Configuration”](#) for details).

The oscillator circuit should be placed on the same side of the board as the device. Also, place the oscillator circuit close to the respective oscillator pins, not exceeding one-half inch (12 mm) distance between them. The load capacitors should be placed next to the oscillator itself, on the same side of the board. Use a grounded copper pour around the oscillator circuit to isolate them from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed. A suggested layout is illustrated in [Figure 2-3](#).

**FIGURE 2-3: SUGGESTED OSCILLATOR CIRCUIT PLACEMENT**



# PIC32MK GPG/MCJ with CAN FD Family

## 2.7.1 CRYSTAL OSCILLATOR DESIGN CONSIDERATION

The following example assumptions are used to calculate the Primary Oscillator loading capacitor values:

- $C_{IN}$  = PIC32\_OSCO\_Pin Capacitance = 4 pF
- $C_{OUT}$  = PIC32\_OSCI\_Pin Capacitance = 4 pF
- PCB stray capacitance (i.e., 12 mm length) = 2.5 pF
- $C_1$  and  $C_2$  = the loading capacitors to use on your crystal circuit design to guarantee that the effective capacitance as seen by the crystal in circuit meets the crystal manufacturer specification

MFG Crystal Data Sheet CLOAD spec:

$$C_{LOAD} = \{ ([C_{IN} + C_1] * [C_{OUT} + C_2]) / [C_{IN} + C_1 + C_2 + C_{OUT}] \} + \text{oscillator PCB stray capacitance}$$

### EXAMPLE 2-1: CRYSTAL LOAD CAPACITOR CALCULATION

Crystal manufacturer data sheet spec example:  $C_{LOAD} = 15 \text{ pF}$

Therefore:

$$MFG \ C_{LOAD} = \{ ([C_{IN} + C_1] * [C_{OUT} + C_2]) / [C_{IN} + C_1 + C_2 + C_{OUT}] \} + \text{estimated oscillator PCB stray capacitance}$$

Assuming  $C_1 = C_2$  and PIC32  $C_{in} = C_{out}$ , the formula can be further simplified and restated to solve for  $C_1$  and  $C_2$  by:

$$\begin{aligned} C_1 = C_2 &= ((2 * MFG \ Cload \ spec) - C_{in} - (2 * PCB \ capacitance)) \\ &= ((2 * 15) - 4 - (2 * 2.5 \ pF)) \\ &= (30 - 4 - 5) \\ &= 21 \ pF \end{aligned}$$

Therefore:

$C_1 = C_2 = 21 \text{ pF}$  is the correct loading capacitors to use on your crystal circuit design to guarantee that the effective capacitance as seen by the crystal in circuit in this example is 15 pF to meet the crystal manufacturer specification.

Tips to increase oscillator gain, (that is, to increase peak-to-peak oscillator signal):

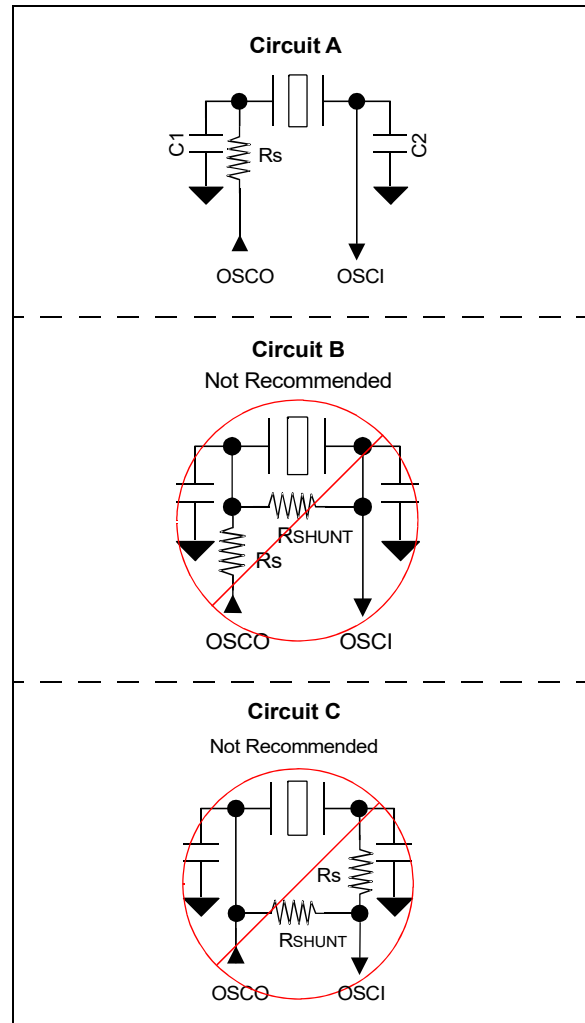
- Select an crystal oscillator with a lower XTAL manufacturing “ESR” rating.
- $C_1$  and  $C_2$  values also affect the gain of the oscillator. The lower the values, the higher the gain.
- Likewise,  $C_2/C_1$  ratio also affects gain. To increase the gain, make  $C_1$  slightly smaller than  $C_2$ , which will also help start-up performance.

**Note:** Do not add excessive gain such that the oscillator signal is clipped, flat on top of the sine wave. If so, you need to reduce the gain or add a series resistor,  $R_s$ , as shown in circuit “A” in Figure 2-4. Failure to do so will stress and age the crystal, which can result in an early failure. When measuring the oscillator signal you must use an active-powered scope probe with  $\leq 1 \text{ pF}$  or the scope probe itself will unduly change the gain and peak-to-peak levels.

### 2.7.1.1 Additional Microchip References

- AN588 “PICmicro® Microcontroller Oscillator Design Guide”
- AN826 “Crystal Oscillator Basics and Crystal Selection for rPIC™ and PICmicro® Devices”
- AN849 “Basic PICmicro® Oscillator Design”

**FIGURE 2-4: PRIMARY CRYSTAL OSCILLATOR CIRCUIT RECOMMENDATIONS**





# PIC32MK GPG/MCJ with CAN FD Family

## 2.8 Unused I/Os

Unused I/O pins should not be allowed to float as inputs. They can be configured as outputs and driven to a logic-low state.

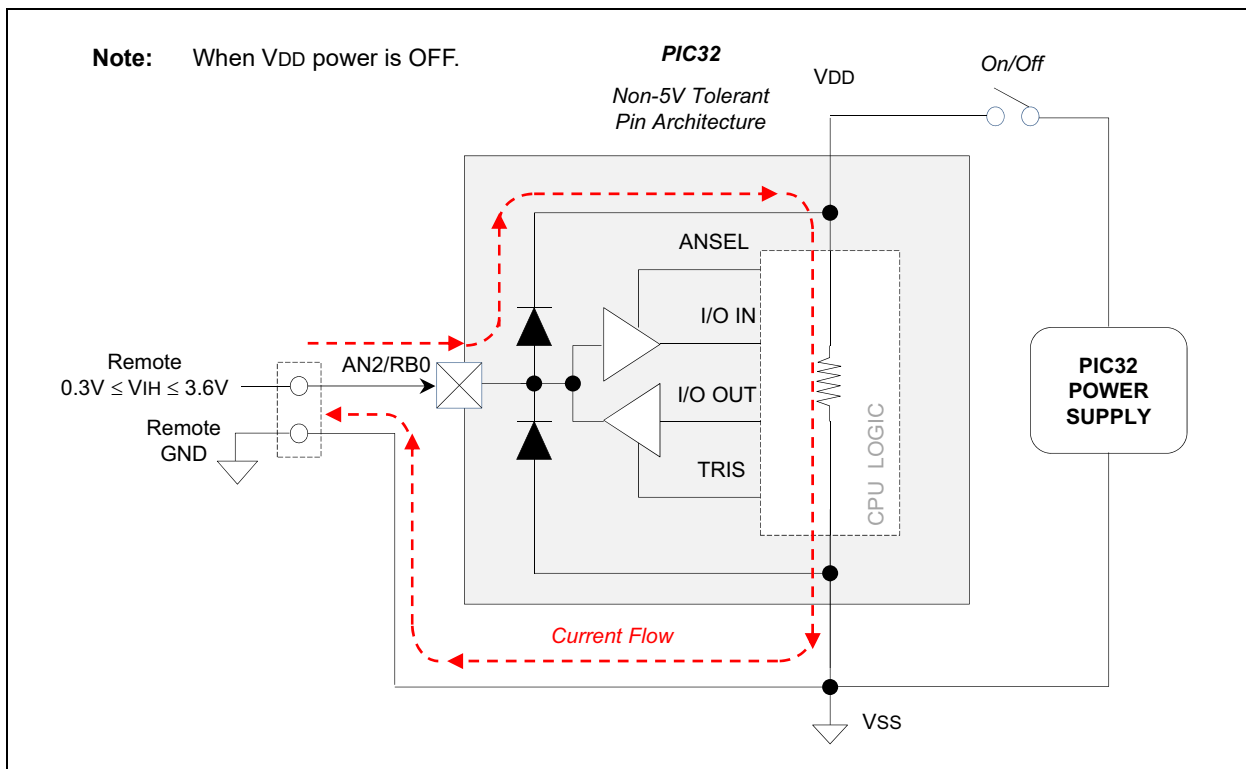
Alternatively, inputs can be reserved by connecting the pin to VSS through a 1k resistor and configuring the pin as an input, which they are by default on any Reset.

## 2.9 Considerations When Interfacing to Remotely Powered Circuits

### 2.9.1 NON-5V TOLERANT INPUT PINS

A quick review of the absolute maximum rating section in **36.0 “Electrical Characteristics”** will indicate that the voltage on any non-5v tolerant pin may not exceed  $V_{DD} + 0.3V$  unless the input current is limited to meet the respective injection current specifications defined by parameters DI60a, DI60b, and DI60c in **TABLE 36-10: “DC Characteristics: I/O Pin Input Injection current Specifications”**. Figure 2-5 shows an example of a remote circuit using an independent power source, which is powered while connected to a PIC32 non-5V tolerant circuit that is not powered.

**FIGURE 2-5: PIC32 NON-5V TOLERANT CIRCUIT EXAMPLE**



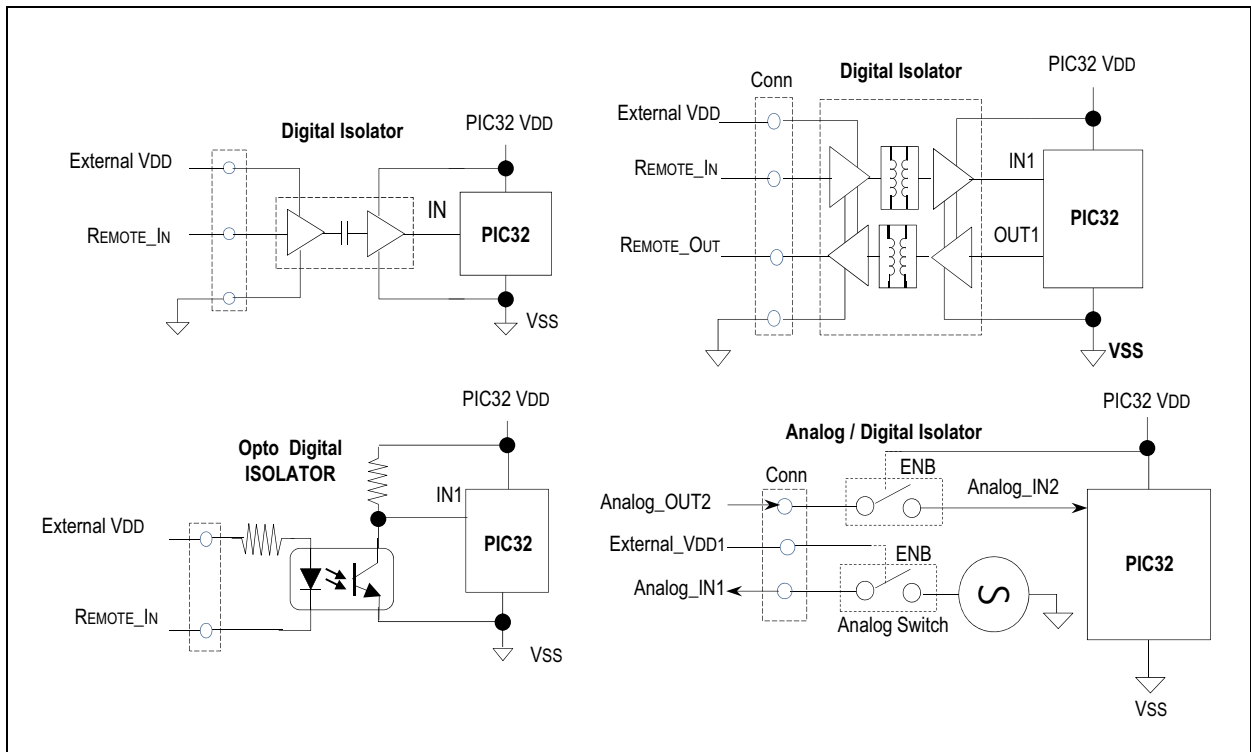
# PIC32MK GPG/MCJ with CAN FD Family

Without proper signal isolation, on non-5V tolerant pins, the remote signal can power the PIC32 device through the high side ESD protection diodes. Besides violating the absolute maximum rating specification when VDD of the PIC32 device is restored and ramping up or ramping down, it can also negatively affect the internal Power-on Reset (POR) and Brown-out Reset (BOR) circuits, which can lead to improper initialization of internal PIC32 logic circuits. In these cases, it is recommended to implement digital or analog signal isolation as depicted in Figure 2-6, as appropriate. This is indicative of all industry microcontrollers and not just Microchip products.

**TABLE 2-1: EXAMPLES OF DIGITAL/ANALOG ISOLATORS WITH OPTIONAL LEVEL TRANSLATION**

| Example Digital/Analog Signal Isolation Circuits | Inductive Coupling | Capacitive Coupling | Opto Coupling | Analog/Digital Switch |
|--|--------------------|---------------------|---------------|-----------------------|
| ADuM7241 / 40 ARZ (1 Mbps)                       | X                  | —                   | —             | —                     |
| ADuM7241 / 40 CRZ (25 Mbps)                      | X                  | —                   | —             | —                     |
| ISO721   | —                  | X                   | —             | —                     |
| LTV-829S (2 Channel)                             | —                  | —                   | X             | —                     |
| LTV-849S (4 Channel)                             | —                  | —                   | X             | —                     |
| FSA266 / NC7WB66                                 | —                  | —                   | —             | X                     |

**FIGURE 2-6: EXAMPLE DIGITAL/ANALOG SIGNAL ISOLATION CIRCUITS**

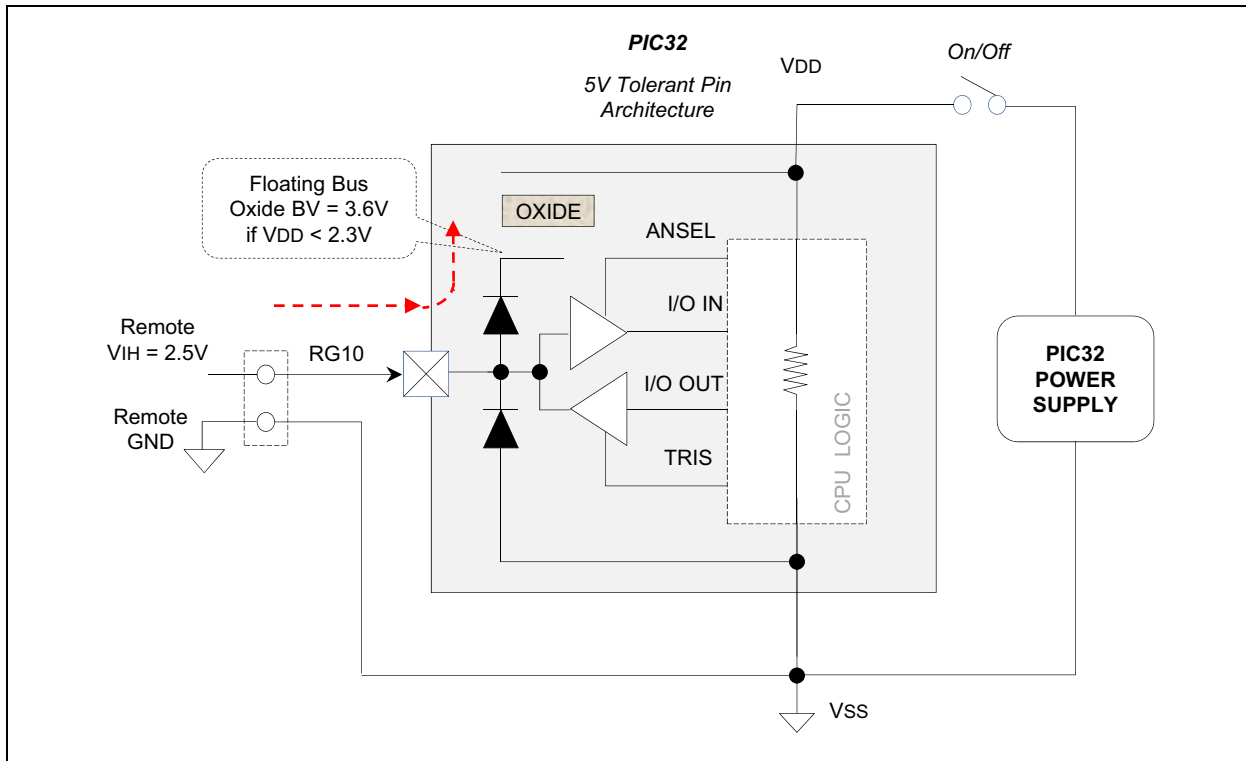


# PIC32MK GPG/MCJ with CAN FD Family

## 2.9.2 5V TOLERANT INPUT PINS

The internal high side diode on 5v tolerant pins are bussed to an internal floating node, rather than being connected to VDD, as shown in Figure 2-7. Voltages on these pins, if  $V_{DD} < 2.3V$ , should not exceed roughly 3.2V relative to Vss of the PIC32 device. Voltage of 3.6V or higher will violate the absolute maximum specification, and will stress the oxide layer separating the high side floating node, which impacts device reliability. If a remotely powered “digital-only” signal can be guaranteed to always be  $\leq 3.2V$  relative to Vss on the PIC32 device side, a 5V tolerant pin could be used without the need for a digital isolator. This is assuming there is not a ground loop issue, logic ground of the two circuits not at the same absolute level, and a remote logic low input is not less than  $V_{ss} - 0.3V$ .

**FIGURE 2-7: PIC32 5V TOLERANT PIN ARCHITECTURE EXAMPLE**



# PIC32MK GPG/MCJ with CAN FD Family

## 2.10 Designing for High-Speed Peripherals

The PIC32MK GPG/MCJ with CAN FD Family of devices have peripherals that operate at frequencies much higher than typical for an embedded environment. Table 2-2 lists the peripherals that produce high-speed signals on their external pins:

**TABLE 2-2: PERIPHERALS THAT PRODUCE HS SIGNALS ON EXTERNAL PINS**

| Peripheral           | High-Speed Signal Pins | Maximum Speed on Signal Pin |
|----------------------|------------------------|-----------------------------|
| SPI/I <sup>2</sup> S | SCKx, SDOx, SDIx       | 50 MHz                      |
| REFCLKx              | REFCLKx                | 50 MHz                      |

Due to these high-speed signals, it is important to consider several factors when designing a product that uses these peripherals, as well as the PCB on which these components will be placed. Adhering to these recommendations will help achieve the following goals:

- Minimize the effects of electromagnetic interference to the proper operation of the product
- Ensure signals arrive at their intended destination at the same time
- Minimize crosstalk
- Maintain signal integrity
- Reduce system noise
- Minimize ground bounce and power sag

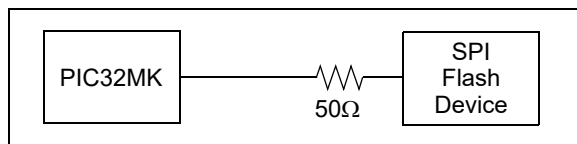
### 2.10.1 SYSTEM DESIGN

#### 2.10.1.1 Impedance Matching

When selecting parts to place on high-speed buses, particularly the SPI bus and REFCLKx outputs, if the impedance of the peripheral device does not match the impedance of the pins on the PIC32MK GPG/MCJ with CAN FD Family of devices to which it is connected, signal reflections could result, thereby degrading the quality of the signal.

If it is not possible to select a product that matches impedance, place a series resistor at the load to create the matching impedance. See Figure 2-8 for an example.

**FIGURE 2-8: SERIES RESISTOR**



#### 2.10.1.2 PCB Layout Recommendations

The following list contains recommendations that will help ensure the PCB layout will promote the goals previously listed.

##### • Component Placement

- Place bypass capacitors as close to their component power and ground pins as possible, and place them on the same side of the PCB.
- Devices on the same bus that have larger setup times must be placed closer to the PIC32MK GPG/MCJ with CAN FD Family of devices.

##### • Power and Ground

- Multi-layer PCBs will allow separate power and ground planes
- Each ground pin should be connected to the ground plane individually
- Place bypass capacitor vias as close to the pad as possible (preferably inside the pad)
- If power and ground planes are not used, maximize width for power and ground traces
- Use low-ESR, surface-mount bypass capacitors

##### • Clocks and Oscillators

- Place crystals as close as possible to the PIC32MK GPG/MCJ with CAN FD Family device OSC/SOSC pins
- Do not route high-speed signals near the clock or oscillator
- Avoid via usage and branches in clock lines (SCK)
- Place termination resistors at the end of clock lines

##### • Traces

- Higher-priority signals should have the shortest traces
- Avoid long run lengths on parallel traces to reduce coupling
- Make the clock traces as straight as possible
- Use rounded turns rather than right-angle turns
- Have traces on different layers intersect on right angles to minimize crosstalk
- Maximize the distance between traces, preferably no less than three times the trace width
- Power traces should be as short and as wide as possible
- High-speed traces should be placed close to the ground plane

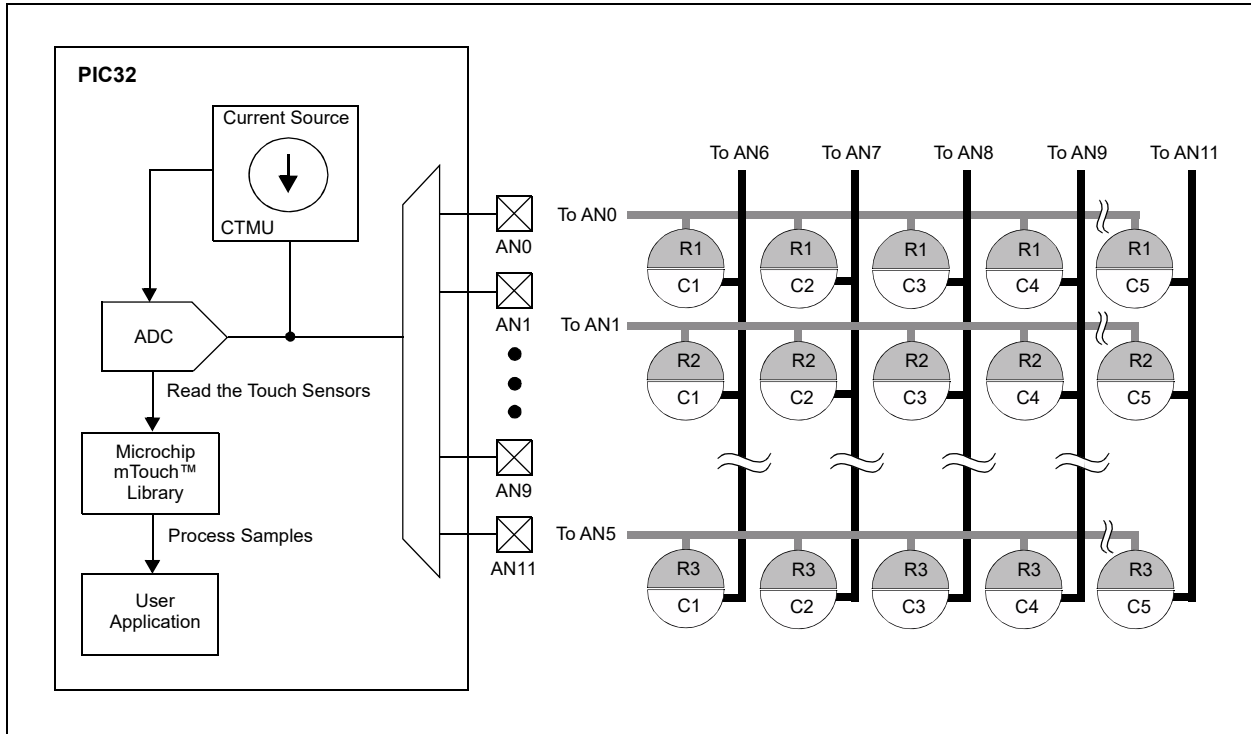


# PIC32MK GPG/MCJ with CAN FD Family

## 2.11 Typical Application Connection Examples

Examples of typical application connections are shown in [Figure 2-10](#), [Figure](#) , and [Figure](#) .

**FIGURE 2-10: CAPACITIVE TOUCH SENSING APPLICATION**



# PIC32MK GPG/MCJ with CAN FD Family

## 2.0 CPU

**Note 1:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 50. “CPU for Devices with MIPS32® microAptiv™ and M-Class Cores”** (DS60001192) of the “*PIC32 Family Reference Manual*”, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

**2:** The microAptiv™ CPU core resources are available at: [www.imgtec.com](http://www.imgtec.com).

The MIPS32® microAptiv™ MCU Core is the heart of the PIC32MK GPG/MCJ with CAN FD Family of device's processor. The CPU fetches instructions, decodes each instruction, fetches source operands, executes each instruction and writes the results of instruction execution to the proper destinations.

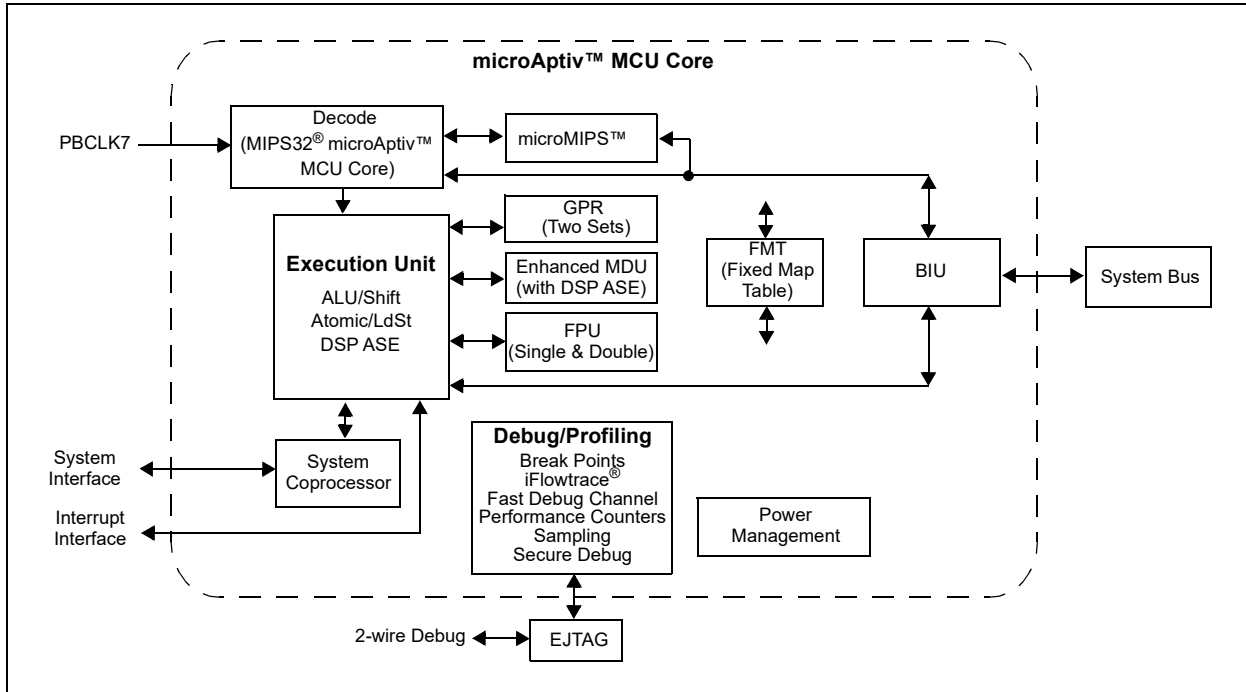
The following are key features of the CPU module:

- 5-stage pipeline
- 32-bit address and data paths
- MIPS32 Enhanced Architecture (Release 5):
  - Multiply-accumulate and multiply-subtract instructions
  - Targeted multiply instruction
  - Zero/One detect instructions
  - WAIT instruction
  - Conditional move instructions (MOVN, MOVZ)
  - Vectored interrupts
  - Programmable exception vector base
  - Atomic interrupt enable/disable
  - GPR shadow registers to minimize latency for interrupt handlers
  - Bit field manipulation instructions
  - Virtual memory support
- microMIPS™ compatible instruction set:
  - Improves code size density over MIPS32, while maintaining MIPS32 performance.
  - Supports all MIPS32 instructions (except branch-likely instructions)
  - Fifteen additional 32-bit instructions and 39 16-bit instructions corresponding to commonly-used MIPS32 instructions
  - Stack pointer implicit in instruction
  - MIPS32 assembly and ABI compatible
- Autonomous Multiply/Divide Unit (MDU):
  - Maximum issue rate of one 32x32 multiply per clock
  - Early-in iterative divide. Minimum 12 and maximum 38 clock latency (dividend (rs) sign extension-dependent)
- Power Control:
  - Minimum frequency: 0 MHz
  - Low-Power mode (triggered by WAIT instruction)
  - Extensive use of local gated clocks
- EJTAG Debug and Instruction Trace:
  - Support for single stepping
  - Virtual instruction and data address/value breakpoints
  - Hardware breakpoint supports both address match and address range triggering.
  - Eight instruction and four data complex breakpoints
- iFlowtrace® version 2.0 support:
  - Real-time instruction program counter
  - Special events trace capability
  - Two performance counters with 34 user-selectable countable events
  - Disabled if the processor enters Debug mode
  - Program Counter sampling
- DSP ASE Extension:
  - Native fractional format data type operations
  - Register Single Instruction Multiple Data (SIMD) operations (add, subtract, multiply, shift)
  - GPR-based shift
  - Bit manipulation
  - Compare-Pick
  - DSP Control Access
  - Indexed-Load
  - Branch
  - Multiplication of complex operands
  - Variable bit insertion and extraction
  - Virtual circular buffers
  - Arithmetic saturation and overflow handling
  - Zero-cycle overhead saturation and rounding operations
- Floating Point Unit (FPU):
  - 1985 IEEE-754 compliant Floating Point Unit
  - Supports single and double precision datatypes
  - 2008 IEEE-754 compatibility control of NaN handling and Abs/Neg instructions
  - Runs at 1:1 core/FPU clock ratio

# PIC32MK GPG/MCJ with CAN FD Family

A typical block diagram of the PIC32MK GPG/MCJ with CAN FD Family processor core is shown in [Figure 2-1](#).

**FIGURE 2-1: PIC32MK GPG/MCJ WITH CAN FD FAMILY MICROPROCESSOR CORE BLOCK DIAGRAM**





# PIC32MK GPG/MCJ with CAN FD Family

## 2.1 Architecture Overview

The MIPS32 microAptiv MCU core in the PIC32MK GPG/MCJ with CAN FD Family of devices contains several logic blocks working together in parallel, providing an efficient high-performance computing engine. The following blocks are included with the core:

- Execution unit
- General Purpose Register (GPR)
- Multiply/Divide Unit (MDU)
- System control coprocessor (CP0)
- Floating Point Unit (FPU)
- Power Management
- microMIPS support
- Enhanced JTAG (EJTAG) controller

### 2.1.1 EXECUTION UNIT

The processor core execution unit implements a load/store architecture with single-cycle ALU operations (logical, shift, add, subtract) and an autonomous multiply/divide unit. The core contains thirty-two 32-bit General Purpose Registers (GPRs) used for integer operations and address calculation. Seven additional register file shadow sets (containing thirty-two registers) are added to minimize context switching overhead during interrupt/exception processing. The register file consists of two read ports and one write port and is fully bypassed to minimize operation latency in the pipeline.

The execution unit includes:

- 32-bit adder used for calculating the data address
- Address unit for calculating the next instruction address
- Logic for branch determination and branch target address calculation
- Load aligner
- Trap condition comparator
- Bypass multiplexers used to avoid stalls when executing instruction streams where data producing instructions are followed closely by consumers of their results

- Leading Zero/One detect unit for implementing the CLZ and CLO instructions
- Arithmetic Logic Unit (ALU) for performing arithmetic and bitwise logical operations
- Shifter and store aligner
- DSP ALU and logic block for performing DSP instructions, such as arithmetic/shift/compare operations

### 2.1.2 MULTIPLY/DIVIDE UNIT (MDU)

The processor core includes a Multiply/Divide Unit (MDU) that contains a separate pipeline for multiply and divide operations, and DSP ASE multiply instructions. This pipeline operates in parallel with the Integer Unit (IU) pipeline and does not stall when the IU pipeline stalls. This allows MDU operations to be partially masked by system stalls and/or other integer unit instructions.

The high-performance MDU consists of a 32x16 Booth recoded multiplier, a pair of result/accumulation registers (HI and LO), a divide state machine, and the necessary multiplexers and control logic. The first number shown ('32' of 32x16) represents the *rs* operand. The second number '16' of 32x16) represents the *rt* operand.

The MDU supports execution of one multiply or multiply-accumulate operation every clock cycle.

Divide operations are implemented with a simple 1-bit-per-clock iterative algorithm. An early-in detection checks the sign extension of the dividend (*rs*) operand. If *rs* is 8 bits wide, 23 iterations are skipped. For a 16-bit wide *rs*, 15 iterations are skipped and for a 24-bit wide *rs*, 7 iterations are skipped. Any attempt to issue a subsequent MDU instruction while a divide is still active causes an IU pipeline stall until the divide operation has completed.

Table 2-1 lists the repeat rate (peak issue rate of cycles until the operation can be reissued) and latency (number of cycles until a result is available) for the processor core multiply and divide instructions. The approximate latency and repeat rates are listed in terms of pipeline clocks.

**TABLE 2-1: MIPS32® microAptiv™ MCU CORE HIGH-PERFORMANCE INTEGER MULTIPLY/DIVIDE UNIT LATENCIES AND REPEAT RATES**

| Opcode   | Operand Size (mul <i>rt</i> ) (div <i>rs</i> ) | Latency | Repeat Rate |
|--|--|---------|-------------|
| MULT/MULTU, MADD/MADDU, MSUB/MSUBU (HI/LO destination) | 16 bits  | 5       | 1           |
|  | 32 bits  | 5       | 1           |
| MUL (GPR destination)                                  | 16 bits  | 5       | 1           |
|  | 32 bits  | 5       | 1           |
| DIV/DIVU   | 8 bits   | 12/14   | 12/14       |
|  | 16 bits  | 20/22   | 20/22       |
|  | 24 bits  | 28/30   | 28/30       |
|  | 32 bits  | 36/38   | 36/38       |

# PIC32MK GPG/MCJ with CAN FD Family

The MIPS architecture defines that the result of a multiply or divide operation be placed in one of four pairs of HI and LO registers. Using the Move-From-HI (MFHI) and Move-From-LO (MFLO) instructions, these values can be transferred to the General Purpose Register file.

In addition to the HI/LO targeted operations, the MIPS32 architecture also defines a multiply instruction, MUL, which places the least significant results in the primary register file instead of the HI/LO register pair. By avoiding the explicit MFLO instruction required when using the LO register, and by supporting multiple destination registers, the throughput of multiply-intensive operations is increased.

Two other instructions, Multiply-Add (MADD) and Multiply-Subtract (MSUB), are used to perform the multiply-accumulate and multiply-subtract operations. The MADD instruction multiplies two numbers and then adds the product to the current contents of the HI and LO registers. Similarly, the MSUB instruction multiplies two operands and then subtracts the product from the HI and LO registers. The MADD and MSUB operations are commonly used in DSP algorithms.

The MDU also implements various shift instructions operating on the HI/LO register and multiply instructions as defined in the DSP ASE. The MDU supports all of the data types required for this purpose and includes three extra HI/LO registers as defined by the ASE.

Table 2-2 lists the latencies and repeat rates for the DSP multiply and dot-product operations. The approximate latencies and repeat rates are listed in terms of pipeline clocks.

**TABLE 2-2: DSP-RELATED LATENCIES AND REPEAT RATES**

| Op code  | Latency | Repeat Rate |
|--|---------|-------------|
| Multiply and dot-product without saturation after accumulation | 5       | 1           |
| Multiply and dot-product with saturation after accumulation    | 5       | 1           |
| Multiply without accumulation                                  | 5       | 1           |

## 2.1.3 SYSTEM CONTROL COPROCESSOR (CP0)

In the MIPS architecture, CP0 is responsible for the virtual-to-physical address translation, the exception control system, the processor's diagnostics capability, the operating modes (Kernel, User and Debug) and whether interrupts are enabled or disabled. Configuration information, such as the presence of options like microMIPS is also available by accessing the CP0 registers, listed in Table 2-3.

**TABLE 2-3: COPROCESSOR 0 REGISTERS**

| Register Number | Register Name | Function  |
|-----------------|---------------|---|
| 0-6             | Reserved      | Reserved in the PIC32MK GP Family core.   |
| 7               | HWREna        | Enables access via the RDHWR instruction to selected hardware registers in Non-privileged mode.   |
| 8               | BadVAddr      | Reports the address for the most recent address-related exception.  |
|                 | BadInstr      | Reports the instruction that caused the most recent exception.  |
|                 | BadInstrP     | Reports the branch instruction if a delay slot caused the most recent exception.  |
| 9               | Count         | Processor cycle count.  |
| 10              | Reserved      | Reserved in the PIC32MK GP Family core.   |
| 11              | Compare       | Core timer interrupt control.   |
| 12              | Status        | Processor status and control.   |
|                 | IntCtl        | Interrupt control of vector spacing.  |
|                 | SRSCtl        | Shadow register set control.  |
|                 | SRSMap        | Shadow register mapping control.  |
|                 | View_IPL      | Allows the Priority Level to be read/written without extracting or inserting that bit from/to the Status register.                        |
|                 | SRSMap2       | Contains two 4-bit fields that provide the mapping from a vector number to the shadow set number to use when servicing such an interrupt. |
| 13              | Cause         | Describes the cause of the last exception.  |
|                 | NestedExc     | Contains the error and exception level status bit values that existed prior to the current exception.                                     |
|                 | View_RIPL     | Enables read access to the RIPL bit that is available in the Cause register.  |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 2-3: COPROCESSOR 0 REGISTERS (CONTINUED)**

| Register Number | Register Name  | Function  |
|-----------------|----------------|---|
| 14              | EPC            | Program counter at last exception.  |
|                 | NestedEPC      | Contains the exception program counter that existed prior to the current exception. |
| 15              | PRID           | Processor identification and revision   |
|                 | Ebase          | Exception base address of exception vectors.  |
|                 | CDMMBase       | Common device memory map base.  |
| 16              | Config         | Configuration register.   |
|                 | Config1        | Configuration register 1.   |
|                 | Config2        | Configuration register 2.   |
|                 | Config3        | Configuration register 3.   |
|                 | Config4        | Configuration register 4.   |
|                 | Config5        | Configuration register 5.   |
|                 | Config7        | Configuration register 7.   |
| 17              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 18              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 19              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 20-22           | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 23              | Debug          | EJTAG debug register.   |
|                 | TraceControl   | EJTAG trace control.  |
|                 | TraceControl2  | EJTAG trace control 2.  |
|                 | UserTraceData1 | EJTAG user trace data 1 register.   |
|                 | TraceBPC       | EJTAG trace breakpoint register.  |
|                 | Debug2         | Debug control/exception status 1.   |
| 24              | DEPC           | Program counter at last debug exception.  |
|                 | UserTraceData2 | EJTAG user trace data 2 register.   |
| 25              | PerfCtl0       | Performance counter 0 control.  |
|                 | PerfCnt0       | Performance counter 0.  |
|                 | PerfCtl1       | Performance counter 1 control.  |
|                 | PerfCnt1       | Performance counter 1.  |
| 26              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 27              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 28              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 29              | Reserved       | Reserved in the PIC32MK GP Family core.   |
| 30              | ErrorEPC       | Program counter at last error exception.  |
| 31              | DeSave         | Debug exception save.   |

# PIC32MK GPG/MCJ with CAN FD Family

## 2.1.4 FLOATING POINT UNIT (FPU)

The Floating Point Unit (FPU), Coprocessor (CP1), implements the MIPS Instruction Set Architecture for floating point computation. The implementation supports the ANSI/IEEE Standard 754 (IEEE for Binary Floating Point Arithmetic) for single- and double-precision data formats. The FPU can be programmed to have thirty-two 32-bit or 64-bit floating point registers used for floating point operations.

The performance is optimized for single precision formats. Most instructions have one FPU cycle throughput and four FPU cycle latency. The FPU implements the multiply-add (MADD) and multiply-sub (MSUB) instructions with intermediate rounding after the multiply function. The result is guaranteed to be the same as executing a MUL and an ADD instruction separately, but the instruction latency, instruction fetch, dispatch bandwidth, and the total number of register accesses are improved.

IEEE denormalized input operands and results are supported by hardware for some instructions. IEEE denormalized results are not supported by hardware in general, but a fast flush-to-zero mode is provided to optimize performance. The fast flush-to-zero mode is enabled through the FCCR register, and use of this mode is recommended for best performance when denormalized results are generated.

The FPU has a separate pipeline for floating point instruction execution. This pipeline operates in parallel with the integer core pipeline and does not stall when the integer pipeline stalls. This allows long-running FPU operations, such as divide or square root, to be partially masked by system stalls and/or other integer unit instructions. Arithmetic instructions are always dispatched and completed in order, but loads and stores can complete out of order. The exception model is "precise" at all times.

Table 2-4 contains the floating point instruction latencies and repeat rates for the processor core. In this table, 'Latency' refers to the number of FPU cycles necessary for the first instruction to produce the result needed by the second instruction. The "Repeat Rate" refers to the maximum rate at which an instruction can be executed per FPU cycle.

**TABLE 2-4: FPU INSTRUCTION LATENCIES AND REPEAT RATES**

| Op code  | Latency (FPU Cycles) | Repeat Rate (FPU Cycles) |
|--|----------------------|--------------------------|
| ABS.[S,D], NEG.[S,D],<br>ADD.[S,D], SUB.[S,D],<br>C.cond.[S,D], MUL.S  | 4                    | 1                        |
| MADD.S, MSUB.S,<br>NMADD.S, NMSUB.S,<br>CABS.cond.[S,D]  | 4                    | 1                        |
| CVT.D.S, CVT.PS.PW,<br>CVT.[S,D].[W,L]   | 4                    | 1                        |
| CVT.S.D,<br>CVT.[W,L].[S,D],<br>CEIL.[W,L].[S,D],<br>FLOOR.[W,L].[S,D],<br>ROUND.[W,L].[S,D],<br>TRUNC.[W,L].[S,D] | 4                    | 1                        |
| MOV.[S,D], MOVE.[S,D],<br>MOVN.[S,D],<br>MOVT.[S,D], MOVZ.[S,D]  | 4                    | 1                        |
| MUL.D  | 5                    | 2                        |
| MADD.D, MSUB.D,<br>NMADD.D, NMSUB.D  | 5                    | 2                        |
| RECIP.S  | 13                   | 10                       |
| RECIP.D  | 26                   | 21                       |
| RSQRT.S  | 17                   | 14                       |
| RSQRT.D  | 36                   | 31                       |
| DIV.S, SQRT.S  | 17                   | 14                       |
| DIV.D, SQRT.D  | 32                   | 29                       |
| MTC1, DMTC1, LWC1,<br>LDC1, LDXC1, LUXC1,<br>LWXC1   | 4                    | 1                        |
| MFC1, DMFC1, SWC1,<br>SDC1, SDXC1, SUXC1,<br>SWXC1   | 1                    | 1                        |

**Legend:** S = Single D = Double  
W = Word L = Long word

# PIC32MK GPG/MCJ with CAN FD Family

The FPU implements a high-performance 7-stage pipeline:

- Decode, register read and unpack (FR stage)
- Multiply tree - double pumped for double (M1 stage)
- Multiply complete (M2 stage)
- Addition first step (A1 stage)
- Addition second and final step (A2 stage)
- Packing to IEEE format (FP stage)
- Register writeback (FW stage)

The FPU implements a bypass mechanism that allows the result of an operation to be forwarded directly to the instruction that needs it without having to write the result to the FPU register and then read it back.

Table 2-5 lists the Coprocessor 1 Registers for the FPU.

**TABLE 2-5: FPU (CP1) REGISTERS**

| Register Number | Register Name | Function  |
|-----------------|---------------|---|
| 0               | FIR           | Floating Point implementation register. Contains information that identifies the FPU. |
| 25              | FCCR          | Floating Point condition codes register.  |
| 26              | FEXR          | Floating Point exceptions register.   |
| 28              | FENR          | Floating Point enables register.  |
| 31              | FCSR          | Floating Point Control and Status register.   |

## 2.2 Power Management

The processor core offers a number of power management features, including low-power design, active power management and power-down modes of operation. The core is a static design that supports slowing or halting the clocks, which reduces system power consumption during Idle periods.

### 2.2.1 INSTRUCTION-CONTROLLED POWER MANAGEMENT

The mechanism for invoking Power-Down mode is through execution of the `WAIT` instruction. For more information on power management, see [31.0 “Power-Saving Features”](#).

### 2.2.2 LOCAL CLOCK GATING

The majority of the power consumed by the processor core is in the clock tree and clocking registers. The PIC32MK family makes extensive use of local gated-clocks to reduce this dynamic power consumption.

## 2.3 EJTAG Debug Support

The processor core provides for an Enhanced JTAG (EJTAG) interface for use in the software debug of application and kernel code. In addition to standard User mode and Kernel modes of operation, the processor core provides a Debug mode that is entered after a debug exception (derived from a hardware breakpoint, single-step exception, etc.) is taken and continues until a Debug Exception Return (`DERET`) instruction is executed. During this time, the processor executes the debug exception handler routine.

The EJTAG interface operates through the Test Access Port (TAP), a serial communication port used for transferring test data in and out of the core. In addition to the standard JTAG instructions, special instructions defined in the EJTAG specification specify which registers are selected and how they are used.

# PIC32MK GPG/MCJ with CAN FD Family

---

## 2.4 MIPS DSP ASE Extension

The MIPS DSP Application-Specific Extension Revision 2 is an extension to the MIPS32 architecture. This extension comprises new integer instructions and states that include new HI/LO accumulator register pairs and a DSP control register. This extension is crucial in a wide range of DSP, multimedia, and DSP-like algorithms covering Audio and Video processing applications. The extension supports native fractional format data type operations, register Single Instruction Multiple Data (SIMD) operations, such as add, subtract, multiply, and shift. In addition, the extension includes the following features that are essential in making DSP algorithms computationally efficient:

- Support for multiplication of complex operands
- Variable bit insertion and extraction
- Implementation and use of virtual circular buffers
- Arithmetic saturation and overflow handling support
- Zero cycle overhead saturation and rounding operations

## 2.5 microMIPS ISA

The processor core supports the microMIPS ISA, which contains all MIPS32 ISA instructions (except for branch-likely instructions) in a new 32-bit encoding scheme, with some of the commonly used instructions also available in 16-bit encoded format. This ISA improves code density through the additional 16-bit instructions while maintaining a performance similar to MIPS32 mode. In microMIPS mode, 16-bit or 32-bit instructions will be fetched and recoded to legacy MIPS32 instruction opcodes in the pipeline's I stage, so that the processor core can have the same microAptiv MPU microarchitecture. Because the microMIPS instruction stream can be intermixed with 16-bit half-word or 32-bit word size instructions on halfword or word boundaries, additional logic is in place to address the word misalignment issues, thus minimizing performance loss.

# PIC32MK GPG/MCJ with CAN FD Family

## 2.6 MIPS32® microAptiv™ MCU Core Configuration

Register 2-1 through Register 2-5 show the default configuration of the MIPS32 microAptiv MCU core, which is included on the PIC32MK GPG/MCJ with CAN FD Family of devices.

### REGISTER 2-1: CONFIG: CONFIGURATION REGISTER; CP0 REGISTER 16, SELECT 0

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-1            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R-0           |
|           | —              | —              | —              | —              | —              | —              | —             | ISP           |
| 23:16     | R-0            | R-0            | R-1            | R-0            | U-0            | R-1            | R-0           | R-0           |
|           | DSP            | UDI            | SB             | MDU            | —              | MM<1:0>        |               | BM            |
| 15:8      | R-0            | R-0            | R-0            | R-0            | R-0            | R-1            | R-0           | R-1           |
|           | BE             | AT<1:0>        |                | AR<2:0>        |                |                | U-0           | U-0           |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0          | —             | —             |
|           | —              | —              | —              | —              | —              | K0<2:0>        |               |               |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit                   |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31 **Reserved:** This bit is hardwired to '1' to indicate the presence of the Config1 register.

bit 30-25 **Unimplemented:** Read as '0'

bit 24 **ISP:** Instruction Scratch Pad RAM bit  
0 = Instruction Scratch Pad RAM is not implemented

bit 23 **DSP:** Data Scratch Pad RAM bit  
0 = Data Scratch Pad RAM is not implemented

bit 22 **UDI:** User-defined bit  
0 = CorExtend User-Defined Instructions are not implemented

bit 21 **SB:** SimpleBE bit  
1 = Only Simple Byte Enables are allowed on the internal bus interface

bit 20 **MDU:** Multiply/Divide Unit bit  
0 = Fast, high-performance MDU

bit 19 **Unimplemented:** Read as '0'

bit 18-17 **MM<1:0>:** Merge Mode bits  
10 = Merging is allowed

bit 16 **BM:** Burst Mode bit  
0 = Burst order is sequential

bit 15 **BE:** Endian Mode bit  
0 = Little-endian

bit 14-13 **AT<1:0>:** Architecture Type bits  
00 = MIPS32

bit 12-10 **AR<2:0>:** Architecture Revision Level bits  
001 = MIPS32 Release 2

bit 9-3 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 2-1: CONFIG: CONFIGURATION REGISTER; CP0 REGISTER 16, SELECT 0 (CONTINUED)

bit 2-0 **K0<2:0>**: Kseg0 Coherency Algorithm bits

- 000 = Reserved
- 001 = Reserved
- 010 = Instruction Prefetch Uncached (Default)
- 011 = Instruction Prefetch cached (Recommended)
- 100 = Reserved
- 
- 
- 
- 111 = Reserved



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 2-2: CONFIG1: CONFIGURATION REGISTER 1; CP0 REGISTER 16, SELECT 1

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-1            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | U-0           |
|           | MMUSIZE<5:0>   |                |                |                |                |                |               | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | R-1            | R-1            | R-0            | R-1           | R-1           |
|           | —              | —              | —              | PC             | WR             | CA             | EP            | FP            |

|                                    |                      |
|------------------------------------|----------------------|
| <b>Legend:</b>                     | r = Reserved bit     |
| R = Readable bit                   | W = Writable bit     |
| U = Unimplemented bit, read as '0' |                      |
| -n = Value at POR                  | '1' = Bit is set     |
|                                    | '0' = Bit is cleared |
|                                    | x = Bit is unknown   |

bit 31 **Reserved:** This bit is hardwired to a '1' to indicate the presence of the Config2 register.

bit 30-25 **MMUSIZE<5:0>:** MMU Size bits

**Note:** This bit field is read as '0' decimal in the fixed table-based MMU core, as no TLB is present.

bit 24-5 **Unimplemented:** Read as '0'

bit 4 **PC:** Performance Counter bit

1 = The processor core contains Performance Counters

bit 3 **WR:** Watch Register Presence bit

1 = No Watch registers are present

bit 2 **CA:** Code Compression Implemented bit

0 = No MIPS16e<sup>®</sup> present

bit 1 **EP:** EJTAG Present bit

1 = Core implements EJTAG

bit 0 **FP:** Floating Point Unit bit

1 = Floating Point Unit is present

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 2-3: CONFIG3: CONFIGURATION REGISTER 3; CP0 REGISTER 16, SELECT 3

| Bit Range | Bit 31/23/15/7                 | Bit 30/22/14/6   | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0                    |
|-----------|--------------------------------|------------------|----------------|----------------|----------------|----------------|---------------|----------------------------------|
| 31:24     | r-1<br>—                       | U-0<br>—         | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—                         |
| 23:16     | U-0<br>—                       | R-0<br>IPLW<1:0> | R-1            | R-0            | R-0            | R-0            | R-1<br>MCU    | R/W-y<br>ISAONEXC <sup>(1)</sup> |
| 15:8      | R-y<br>ISA<1:0> <sup>(1)</sup> | R-y              | R-1<br>ULRI    | R-1<br>RXI     | R-1<br>DSP2P   | R-1<br>DSPP    | U-0           | R-1<br>ITL                       |
| 7:0       | U-0<br>—                       | R-1<br>VEIC      | R-1<br>VINT    | R-0<br>SP      | R-1<br>CDMM    | U-0            | U-0           | R-0<br>TL                        |

|                   |                  |  |
|-------------------|------------------|--|
| <b>Legend:</b>    | r = Reserved bit | y = Value set from Configuration bits on POR |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

- bit 31 **Reserved:** This bit is hardwired as '1' to indicate the presence of the Config4 register
- bit 30-23 **Unimplemented:** Read as '0'
- bit 22-21 **IPLW<1:0>:** Width of the Status IPL and Cause RIPL bits  
01 = IPL and RIPL bits are 8-bits in width
- bit 20-18 **MMAR<2:0>:** microMIPS Architecture Revision Level bits  
000 = Release 1
- bit 17 **MCU:** MIPS<sup>®</sup> MCU<sup>™</sup> ASE Implemented bit  
1 = MCU ASE is implemented
- bit 16 **ISAONEXC:** ISA on Exception bit<sup>(1)</sup>  
1 = microMIPS is used on entrance to an exception vector  
0 = MIPS32 ISA is used on entrance to an exception vector
- bit 15-14 **ISA<1:0>:** Instruction Set Availability bits<sup>(1)</sup>  
11 = Both MIPS32 and microMIPS are implemented; microMIPS is used when coming out of reset  
10 = Both MIPS32 and microMIPS are implemented; MIPS32 ISA used when coming out of reset
- bit 13 **ULRI:** UserLocal Register Implemented bit  
1 = UserLocal Coprocessor 0 register is implemented
- bit 12 **RXI:** RIE and XIE Implemented in PageGrain bit  
1 = RIE and XIE bits are implemented
- bit 11 **DSP2P:** MIPS DSP ASE Revision 2 Presence bit  
1 = DSP Revision 2 is present
- bit 10 **DSPP:** MIPS DSP ASE Presence bit  
1 = DSP is present
- bit 9 **Unimplemented:** Read as '0'
- bit 8 **ITL:** Indicates that iFlowtrace<sup>®</sup> hardware is present  
1 = The iFlowtrace<sup>®</sup> 2.0 hardware is implemented in the core
- bit 7 **Unimplemented:** Read as '0'
- bit 6 **VEIC:** External Vector Interrupt Controller bit  
1 = Support for an external interrupt controller is implemented.
- bit 5 **VINT:** Vector Interrupt bit  
1 = Vector interrupts are implemented
- bit 4 **SP:** Small Page bit  
0 = 4 KB page size

**Note 1:** These bits are set based on the value of the BOOTISA Configuration bit (DEVCFG0<6>).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 2-3: CONFIG3: CONFIGURATION REGISTER 3; CP0 REGISTER 16, SELECT 3 (CONTINUED)

bit 3 **CDMM:** Common Device Memory Map bit  
1 = CDMM is implemented

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **TL:** Trace Logic bit  
0 = Trace logic is not implemented

**Note 1:** These bits are set based on the value of the BOOTISA Configuration bit (DEVCFG0<6>).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 2-4: CONFIG4: CONFIGURATION REGISTER 4; CP0 REGISTER 16, SELECT 4

| Bit Range | Bit 31/23/15/7  | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-1             | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | M               | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | R-0             | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | KScr Exist<7:0> |                |                |                |                |                |               |               |
| 15:8      | R/W-0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —               | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | R/W-0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —               | —              | —              | —              | —              | —              | —             | —             |

|                   |              |                  |                                    |
|-------------------|--------------|------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved | W = Writable bit | U = Unimplemented bit, read as '0' |
| R = Readable bit  |              | '1' = Bit is set | '0' = Bit is cleared               |
| -n = Value at POR |              |                  | x = Bit is unknown                 |

bit 31 **M:** Config5 Register Present bit

1 = Config5 register is present

0 = Config5 register is not present

bit 30-24 **Unimplemented:** Read as '0'

bit 23-16 **KScr Exist<7:0>:** Number of Scratch Registers Available to Kernel Mode bits

Indicates how many scratch registers are available to Kernel mode software within CP0 Register 31. Each bit represents a select for Coprocessor0 Register 31. Bit 16 represents Select 0. Bit 23 represents Select 7. If the bit is set, the associated scratch register is implemented and is available for Kernel mode software.

**Note:** These bits are read-only, and this field is all zeros on these products, as is read as '0'.

bit 15-0 **Reserved:** Read/write as '0'

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 2-5: CONFIG5: CONFIGURATION REGISTER 5; CP0 REGISTER 16, SELECT 5**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R-1           |
|           | —              | —              | —              | —              | —              | —              | —             | NF            |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved                       |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31-1 **Unimplemented:** Read as '0'  
bit 0 **NF:** Nested Fault bit  
1 = Nested Fault feature is implemented

**REGISTER 2-6: CONFIG7: CONFIGURATION REGISTER 7; CP0 REGISTER 16, SELECT 7**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-1            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | WII            | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | W = Writable bit                   |
| R = Readable bit  | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set                   |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31 **WII:** Wait IE Ignore bit  
1 = Indicates that this processor will allow an interrupt to unblock a WAIT instruction  
bit 30-0 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 2-7: FIR: FLOATING POINT IMPLEMENTATION REGISTER; CP1 REGISTER 0

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R-1            | U-0            | U-0            | U-0           | R-1           |
|           | —              | —              | —              | UFRP           | —              | —              | —             | FC            |
| 23:16     | R-1            | R-1            | R-1            | R-1            | R-0            | R-0            | R-1           | R-1           |
|           | HAS2008        | F64            | L              | W              | MIPS3D         | PS             | D             | S             |
| 15:8      | R-1            | R-0            | R-1            | R-0            | R-0            | R-1            | R-1           | R-1           |
|           | PRID<7:0>      |                |                |                |                |                |               |               |
| 7:0       | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | REVISION<7:0>  |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28 **UFRP:** User Mode FR Switching Instruction bit  
 1 = User mode FR switching instructions are supported  
 0 = User mode FR switching instructions are not supported

bit 27-25 **Unimplemented:** Read as '0'

bit 24 **FC:** Full Convert Ranges bit  
 1 = Full convert ranges are implemented (all numbers can be converted to another type by the FPU)  
 0 = Full convert ranges are not implemented

bit 23 **HAS008:** IEEE-754-2008 bit  
 1 = MAC2008, ABS2008, NAN2008 bits exist within the FCSR register  
 0 = MAC2009, ABS2008, and NAN2008 bits do not exist within the FCSR register

bit 22 **F64:** 64-bit FPU bit  
 1 = This is a 64-bit FPU  
 0 = This is not a 64-bit FPU

bit 21 **L:** Long Fixed Point Data Type bit  
 1 = Long fixed point data types are implemented  
 0 = Long fixed point data types are not implemented

bit 20 **W:** Word Fixed Point data type bit  
 1 = Word fixed point data types are implemented  
 0 = Word fixed point data types are not implemented

bit 19 **MIPS3D:** MIPS-3D ASE bit  
 1 = MIPS-3D is implemented  
 0 = MIPS-3D is not implemented

bit 18 **PS:** Paired Single Floating Point data bit  
 1 = PS floating point is implemented  
 0 = PS floating point is not implemented

bit 17 **D:** Double-precision floating point data bit  
 1 = Double-precision floating point data types are implemented  
 0 = Double-precision floating point data types are not implemented

bit 16 **S:** Single-precision Floating Point Data bit  
 1 = Single-precision floating point data types are implemented  
 0 = Single-precision floating point data types are not implemented

bit 15-8 **PRID<7:0>:** Processor Identification bits  
 These bits allow software to distinguish between the various types of MIPS processors. For PIC32 devices with the MIPS32 microAptiv MCU core, this value is 0x9D.

bit 7-0 **REVISION<7:0>:** Processor Revision Identification bits  
 These bits allow software to distinguish between one revision and another of the same processor type. This number is increased on major revisions of the processor core

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 2-8: FCCR: FLOATING POINT CONDITION CODES REGISTER; CP1 REGISTER 25**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | FCC<7:0>       |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-8 **Unimplemented:** Read as '0'

bit 7-0 **FCC<7:0>:** Floating Point Condition Code bits

These bits record the results of floating point compares and are tested for floating point conditional branches and conditional moves.





# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 2-10: FENR: FLOATING POINT EXCEPTIONS AND MODES ENABLE REGISTER; CP1 REGISTER 28

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | —              | —              | —              | —              | ENABLES<4:1>   |                |               |               |
|           |                |                |                |                | V              | Z              | O             | U             |
| 7:0       | R/W-x          | U-0            | U-0            | U-0            | U-0            | R-x            | R/W-x         | R/W-x         |
|           | ENABLES<0>     | —              | —              | —              | —              | FS             | RM<1:0>       |               |
|           | I              |                |                |                |                |                |               |               |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-12 **Unimplemented:** Read as '0'

bit 11-7 **ENABLES<4:0>:** FPU Exception Enable bits

These bits control whether or not a trap is taken when an IEEE exception condition occurs for any of the five conditions. The trap occurs when both an enable bit and its corresponding cause bit are set either during an FPU arithmetic operation or by moving a value to the FCSR or one of its alternative representations.

bit 11 **V:** Invalid Operation bit

bit 10 **Z:** Divide-by-Zero bit

bit 9 **O:** Overflow bit

bit 8 **U:** Underflow bit

bit 7 **I:** Inexact bit

bit 6-3 **Unimplemented:** Read as '0'

bit 2 **FS:** Flush to Zero control bit

1 = Denormal input operands are flushed to zero. Tiny results are flushed to either zero or the applied format's smallest normalized number (MinNorm) depending on the rounding mode settings.

0 = Denormal input operands result in an Unimplemented Operation exception.

bit 1-0 **RM<1:0>:** Rounding Mode control bits

11 = Round towards Minus Infinity ( $-\infty$ )

10 = Round towards Plus Infinity ( $+\infty$ )

01 = Round toward Zero (0)

00 = Round to Nearest



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 2-11: FCSR: FLOATING POINT CONTROL AND STATUS REGISTER; CP1 REGISTER 31

- bit 14    **O**: Overflow bit
- bit 13    **U**: Underflow bit
- bit 12    **I**: Inexact bit
- bit 11-7    **ENABLES<4:0>**: FPU Exception Enable bits  
These bits control whether or not a trap is taken when an IEEE exception condition occurs for any of the five conditions. The trap occurs when both an enable bit and its corresponding cause bit are set either during an FPU arithmetic operation or by moving a value to the FCSR or one of its alternative representations.
- bit 11    **V**: Invalid Operation bit
- bit 10    **Z**: Divide-by-Zero bit
- bit 9    **O**: Overflow bit
- bit 8    **U**: Underflow bit
- bit 7    **I**: Inexact bit
- bit 6-2    **FLAGS<4:0>**: FPU Flags bits  
These bits show any exception conditions that have occurred for completed instructions since the flag was last reset by software.
- bit 6    **V**: Invalid Operation bit
- bit 5    **Z**: Divide-by-Zero bit
- bit 4    **O**: Overflow bit
- bit 3    **U**: Underflow bit
- bit 2    **I**: Inexact bit
- bit 1-0    **RM<1:0>**: Rounding Mode control bits  
11 = Round towards Minus Infinity ( $-\infty$ )  
10 = Round towards Plus Infinity ( $+\infty$ )  
01 = Round toward Zero (0)  
00 = Round to Nearest

# PIC32MK GPG/MCJ with CAN FD Family

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

---

## 3.0 MEMORY ORGANIZATION

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. For detailed information, refer to **Section 48. “Memory Organization and Permissions”** (DS60001214), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

PIC32MK GPG/MCJ with CAN FD Family microcontrollers provide 4 GB of unified virtual memory address space. All memory regions, including program, data memory, Special Function Registers (SFRs) and Configuration registers, reside in this address space at their respective unique addresses. The program and data memories can be optionally partitioned into user and kernel memories. In addition, PIC32MK GPG/MCJ with CAN FD Family of devices allow execution from data memory.

The following are key features of this module:

- 32-bit native data width
- Separate User (KUSEG) and Kernel (KSEG0/KSEG1) mode address space
- Separate boot Flash memory for protected code
- Robust bus exception handling to intercept runaway code
- Read/write permission access to predefined memory regions

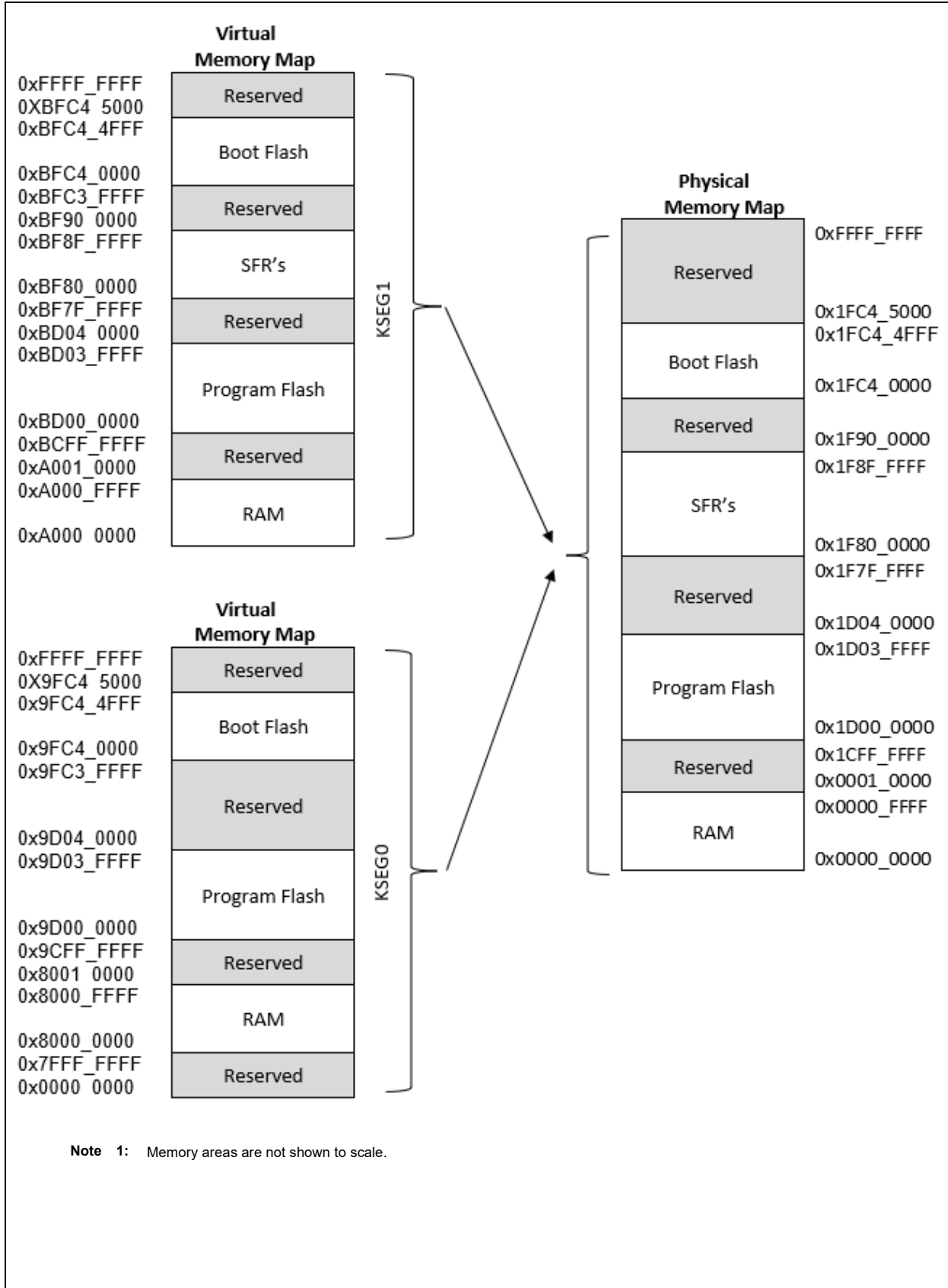
## 3.1 Memory Layout

PIC32MK GPG/MCJ with CAN FD Family microcontrollers implement two address schemes: virtual and physical. All hardware resources, such as program memory, data memory and peripherals, are located at their respective physical addresses. Virtual addresses are exclusively used by the CPU to fetch and execute instructions as well as access peripherals. Physical addresses are used by bus master peripherals, such as DMA and the Flash controller, that access memory independently of the CPU.

The main memory maps for the PIC32MK GPG/MCJ with CAN FD Family of devices are illustrated in [Figure 3-1](#) through [Figure 3-2](#). [Figure 3-3](#) provides memory map information for boot Flash and boot alias. [Table 3-3](#) provides memory map information for SFRs.

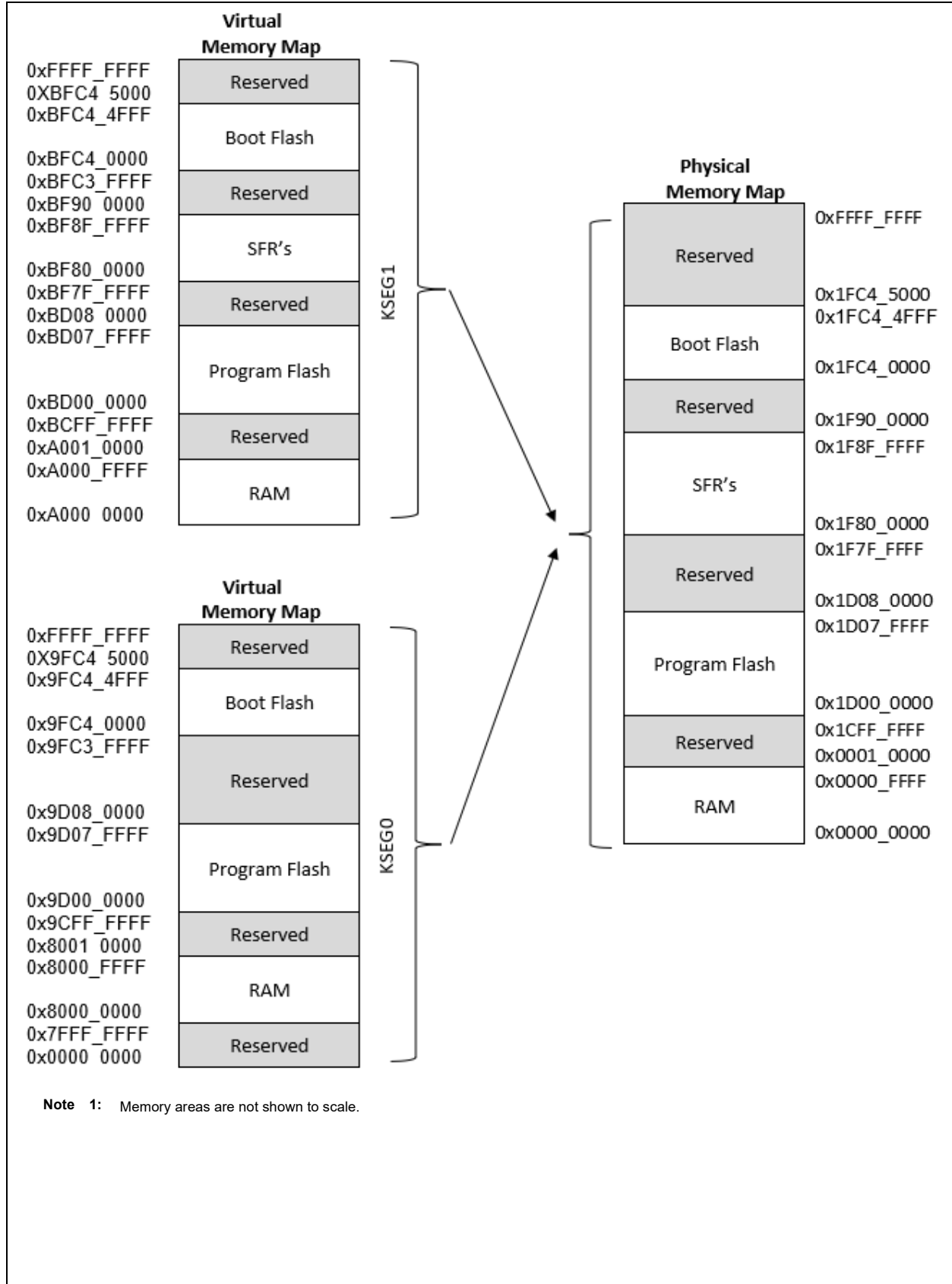
# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 3-1: MEMORY MAP FOR DEVICES WITH 256 KB PROGRAM MEMORY AND 64 KB RAM**



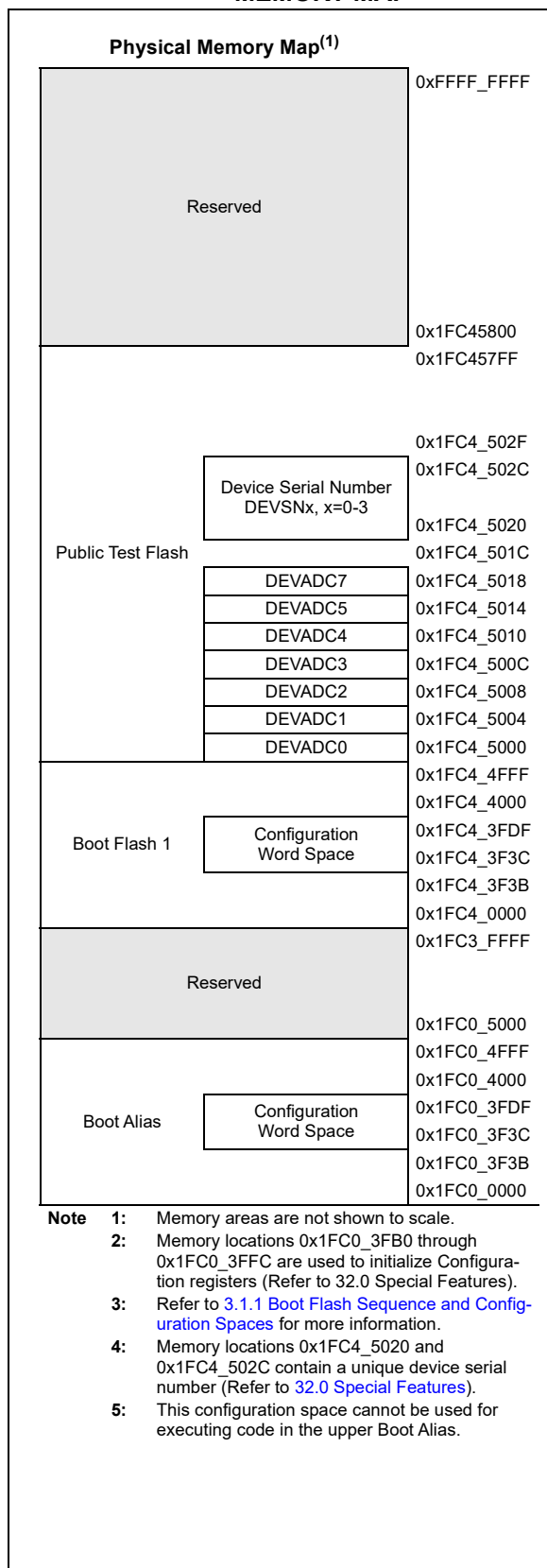
# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 3-2: MEMORY MAP FOR DEVICES WITH 512 KB PROGRAM MEMORY AND 64 KB RAM**



# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 3-3: BOOT AND ALIAS MEMORY MAP**



**TABLE 3-1: SFR MEMORY MAP**

| Peripheral    | Virtual Address |              |
|---------------|-----------------|--------------|
|               | Base            | Offset Start |
| CFG-PMD       | 0xBF800000      | 0x0000       |
| CACHE         |                 | 0x0800       |
| FC-NVM        |                 | 0x0A00       |
| WDT           |                 | 0x0C00       |
| DMT           |                 | 0x0E00       |
| ICD           |                 | 0x1000       |
| CRU           |                 | 0x1200       |
| PPS           |                 | 0x1400       |
| HLVD          |                 | 0x1800       |
| EVIC          |                 | 0xBF810000   |
| DMA           | 0x1000          |              |
| Timer1-Timer9 | 0xBF820000      | 0x0000       |
| IC1-IC9       |                 | 0x2000       |
| OC1-OC9       |                 | 0x4000       |
| I2C1-I2C2     |                 | 0x6000       |
| SPI1-SPI2     |                 | 0x7000       |
| UART1-UART2   |                 | 0x8000       |
| PWM1 - PWM9   |                 | 0xA000       |
| QE11-QE13     |                 | 0xB200       |
| CMP           |                 | 0xC000       |
| CTMU          |                 | 0xD000       |
| CLC1-CLC4     |                 | 0xD200       |
| CDAC1-CDAC2   | 0xBF840000      | 0xC400       |
| PORTA-PORTG   | 0xBF860000      | 0x0000       |
| CANFD1        | 0xBF880000      | 0x0000       |
| ADC           |                 | 0x7000       |
| RTCC          | 0xBF8C0000      | 0x0000       |
| SSX CTL       | 0xBF8F0000      | 0x0000       |

**Note 1:** Refer to 3.2 “System Bus Arbitration” for important legal information.



# PIC32MK GPG/MCJ with CAN FD Family

---

## 3.1.1 BOOT FLASH SEQUENCE AND CONFIGURATION SPACES

Sequence space is used to identify which boot Flash is aliased by aliased regions. If the value programmed into the TSEQ<15:0> bits of the BF1SEQ word is equal to or greater than the value programmed into the TSEQ<15:0> bits of the BF2SEQ word, Boot Flash 1 is aliased by the lower boot alias region, and Boot Flash 2 is aliased by the upper boot alias region. If the TSEQ<15:0> bits of the BF2SEQ word is greater than the TSEQ<15:0> bits of the BF1SEQ word, the opposite is true (see [Table 3-2](#) and [Table 4-3](#) for BFxSEQ word memory locations).

Once boot Flash memories are aliased, configuration space located in the lower boot alias region is used as the basis for the Configuration words, DEVSIGN0, DEVCP0, and DEVCFGx. This means that the boot Flash region to be aliased by lower boot alias region memory must contain configuration values in the appropriate memory locations.

|  |
|--|
| <p><b>Note:</b> Use only Quad Word program operation (NVMOP&lt;3:0&gt; = 0010) when programming data into the sequence and configuration spaces.</p> |
|--|

## 3.1.2 ALTERNATE SEQUENCE AND CONFIGURATION WORDS

Every word in the configuration space and sequence space has an associated alternate word (designated by the letter A as the first letter in the name of the word). During device start-up, primary words are read, and if uncorrectable ECC errors are found, the BCFGERR (RCON<27>) flag is set and alternate words are used. If uncorrectable ECC errors are found in primary and alternate words, the BCFGFAIL (RCON<26>) flag is set and the default configuration is used.

**TABLE 3-2: BOOT FLASH 1 SEQUENCE AND CONFIGURATION WORDS SUMMARY**

| Virtual Address (BFC4_#) | Register Name | Bit Range | Bits   |       |       |       |       |       |      |      |      |      |      |      |      |      | All Reset |
|--------------------------|---------------|-----------|--|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-----------|
|                          |               |           | 31/15  | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |           |
| 3F40                     | ABF1DEVCFG3   | 31:16     | Note: See <a href="#">Table 32-1</a> for the bit descriptions. |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F44                     | ABF1DEVCFG2   | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F48                     | ABF1DEVCFG1   | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F4C                     | ABF1DEVCFG0   | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F50                     | ABF1DEVCP3    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F54                     | ABF1DEVCP2    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F58                     | ABF1DEVCP1    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3F5C                     | ABF1DEVCP0    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3FC0                     | BF1DEVCFG3    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3FC4                     | BF1DEVCFG2    | 31:16     |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
|                          |               | 15:0      |  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx      |
| 3FC8                     | BF1DEVCFG1    | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
| 3FCC                     | BF1DEVCFG0    | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
| 3FD0                     | BF1DEVCP3     | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
| 3FD4                     | BF1DEVCP2     | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
| 3FD8                     | BF1DEVCP1     | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
| 3FDC                     | BF1DEVCP0     | 31:16     | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |
|                          |               | 15:0      | xxxx   |       |       |       |       |       |      |      |      |      |      |      |      |      |           |

**Legend:** x = unknown value on Reset; — = Reserved, read as '1'. Reset values are shown in hexadecimal.

# PIC32MK GPG/MCJ with CAN FD Family

## 3.2 System Bus Arbitration

**Note:** The System Bus interconnect implements one or more instantiations of the SonicsSX<sup>®</sup> interconnect from Sonics, Inc. This document contains materials that are (c) 2003-2015 Sonics, Inc., and that constitute proprietary information of Sonics, Inc. SonicsSX is a registered trademark of Sonics, Inc. All such materials and trademarks are used under license from Sonics, Inc.

As shown in the PIC32MK GPG/MCJ with CAN FD Family Block Diagram (see [Figure 1-1](#)), there are multiple initiator modules (I1 through I13) in the system that can access various target modules (T1 through T14). [Table 3-3](#) illustrates which initiator can access which target. The System Bus supports simultaneous access to targets by initiators, so long as the initiators are accessing different targets. The System Bus will perform arbitration, if multiple initiators attempt to access the same target.

**TABLE 3-3: INITIATORS TO TARGETS ACCESS ASSOCIATION**

| Target # | Initiator ID:  | 1      | 2      | 3        | 4         | 5     | 6        | 7        | 10   |
|----------|--|--------|--------|----------|-----------|-------|----------|----------|------|
|          | Name:  | CPU IS | CPU ID | DMA Read | DMA Write | Flash | ICD JTAG | ADC Mem. | CAN1 |
| 1        | Program Flash  | X      |        | X        |           |       |          |          |      |
| 2        | Data   |        | X      |          |           |       |          |          |      |
| 3        | Peripheral Module  |        |        | X        |           |       | X        |          | X    |
| 4        | RAM Bank 1   | X      | X      | X        | X         | X     | X        | X        | X    |
| 5        | RAM Bank 2   | X      | X      | X        | X         | X     | X        | X        | X    |
| 7        | <b>Peripheral Bus 1:</b><br>DMT, CVR,<br>PPS Input,<br>PPS Output,<br>WDT  |        |        |          |           |       | X        |          |      |
| 8        | <b>Peripheral Bus 2:</b><br>Timer1-Timer9,<br>I2C1-I2C2,<br>SPI1-SPI2,<br>UART1-UART2,<br>OC1-OC9,<br>IC1-IC9,<br>PMP,<br>Comparator 1-<br>Comparator 5,<br>Op amp 1-3,5<br>PWM1-PWM9<br>QE11-QE12 |        | X      | X        | X         |       | X        |          |      |
| 9        | <b>Peripheral Bus 3:</b><br>CDAC1-CDAC2  |        | X      | X        | X         |       | X        |          |      |
| 10       | <b>Peripheral Bus 4:</b><br>PORTA-PORTG  |        | X      | X        | X         |       | X        |          |      |
| 11       | <b>Peripheral Bus 5:</b><br>CAN1<br>ADC 0-5, 7   |        | X      |          |           |       | X        |          |      |
| 14       | <b>Peripheral Bus 6:</b><br>RTCC   |        | X      |          |           |       | X        |          |      |

# PIC32MK GPG/MCJ with CAN FD Family

---

The System Bus arbitration scheme implements a non-programmable, Least Recently Serviced (LRS) priority, which provides Quality Of Service (QOS) for most initiators. However, some initiators can use Fixed High Priority (HIGH) arbitration to guarantee their access to data.

The arbitration scheme for the available initiators is shown in [Table 3-4](#).

**TABLE 3-4: INITIATOR ID AND QOS**

| Name             | ID | QOS  |
|------------------|----|------|
| CPU-IS           | 1  | LRS  |
| CPU-DS           | 2  | LRS  |
| DMA Read         | 3  | LRS  |
| DMA Write        | 4  | LRS  |
| Flash Controller | 5  | HIGH |
| ICD-JTAG         | 6  | LRS  |
| ADC              | 7  | LRS  |
| CAN1             | 10 | LRS  |

## 3.3 Permission Access and System Bus Registers

The System Bus on PIC32MK GPG/MCJ with CAN FD Family of devices provides access control capabilities for the transaction initiators on the System Bus.

The System Bus divides the entire memory space into fourteen target regions and permits access to each target by initiators through permission groups. Four Permission Groups (0 through 3) can be assigned to each initiator. Each permission group is independent of the others and can have exclusive or shared access to a region.

Using the CFGPG register (see [Register 32-8](#) in [32.0 “Special Features”](#)), Boot firmware can assign a permission group to each initiator, which can make requests on the System Bus.

The available targets and their regions, as well as the associated control registers to assign protection, are described and listed in [Table 3-5](#).

[Register 3-1](#) through [Register 3-9](#) are used for setting and controlling access permission groups and regions.

To change these registers, they must be unlocked in hardware. The register lock is controlled by the PGLOCK Configuration bit (CFGCON<11>). Setting the PGLOCK bit prevents writes to the control registers and clearing the PGLOCK bit allows writes.

To set or clear the PGLOCK bit, an unlock sequence must be executed. Refer to [Section 42. “Oscillators with Enhanced PLL”](#) (DS60001250) in the *“PIC32 Family Reference Manual”* for details.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 3-5: SYSTEM BUS TARGETS AND ASSOCIATED PROTECTION REGISTERS**

| Target Number | Target Description   | SBTxREGy Register |          |                        |             |                | SBTxRDy Register |  | SBTxWRy Register |   |
|---------------|--|-------------------|----------|------------------------|-------------|----------------|------------------|--|------------------|---|
|               |  | Name              | Region   | Physical Start Address | Region Size | Priority Level | Name             | Read Permission (Group3, Group2, Group1, Group0) | Name             | Write Permission (Group3, Group2, Group1, Group0) |
| 0             | System Bus   | SBT0REG0          | Region 0 | 1F8F0000               |             | 0              | SBT0RD0          | 1,1,1,1  | SBT0WR0          | 1,1,1,1   |
|               |  | SBT0REG1          | Region 1 | 1F8F8000               | 32 KB       | 3              | SBT0RD1          | 0,0,0,1  | SBT0WR1          | 0,0,0,1   |
| 1             | Flash Memory (CPU Instruction)<br>Program Flash<br>Boot Flash Prefetch | SBT1REG0          | Region 0 | 1D000000               |             | 0              | SBT1RD0          | 1,1,1,1  | SBT1WR0          | 0,0,0,0   |
|               |  | SBT1REG2          | Region 2 | 1FC04000               | 4 KB        | 2              | SBT1RD2          | 0,0,0,1  | SBT1WR2          | 0,0,0,0   |
|               |  | SBT1REG3          | Region 3 | 1FC24000               | 4 KB        | 2              | SBT1RD3          | 0,0,0,1  | SBT1WR3          | 0,0,0,0   |
|               |  | SBT1REG4          | Region 4 | 1FC44000               | 4 KB        | 2              | SBT1RD4          | 0,0,0,1  | SBT1WR4          | 0,0,0,0   |
|               |  | SBT1REG5          | Region 5 | 1FC64000               | 4 KB        | 2              | SBT1RD5          | 0,0,0,1  | SBT1WR5          | 0,0,0,0   |
| 2             | Flash Memory (CPU data)<br>Program Flash                               | SBT2REG0          | Region 0 | 1D000000               |             | 0              | SBT2RD0          | 1,1,1,1  | SBT2WR0          | 0,0,0,0   |
|               |  | SBT2REG2          | Region 2 | 1FC04000               | 4 KB        | 2              | SBT2RD2          | 0,0,0,1  | SBT2WR2          | 0,0,0,0   |
|               |  | SBT2REG3          | Region 3 | 1FC24000               | 4 KB        | 2              | SBT2RD3          | 0,0,0,1  | SBT2WR3          | 0,0,0,0   |
|               |  | SBT2REG4          | Region 4 | 1FC44000               | 4 KB        | 2              | SBT2RD4          | 0,0,0,1  | SBT2WR4          | 0,0,0,0   |
|               |  | SBT2REG5          | Region 5 | 1FC64000               | 4 KB        | 2              | SBT2RD5          | 0,0,0,1  | SBT2WR5          | 0,0,0,0   |
| 3             | Flash Memory (peripheral)<br>Program Flash                             | SBT3REG0          | Region 0 | 1D000000               |             | 0              | SBT3RD0          | 1,1,1,1  | SBT3WR0          | 0,0,0,0   |
|               |  | SBT3REG2          | Region 2 | 1FC04000               | 4 KB        | 2              | SBT3RD2          | 0,0,0,1  | SBT3WR2          | 0,0,0,0   |
|               |  | SBT3REG3          | Region 3 | 1FC24000               | 4 KB        | 2              | SBT3RD3          | 0,0,0,1  | SBT3WR3          | 0,0,0,0   |
|               |  | SBT3REG4          | Region 4 | 1FC44000               | 4 KB        | 2              | SBT3RD4          | 0,0,0,1  | SBT3WR4          | 0,0,0,0   |
|               |  | SBT3REG5          | Region 5 | 1FC64000               | 4 KB        | 2              | SBT3RD5          | 0,0,0,1  | SBT3WR5          | 0,0,0,0   |

**Legend:** R = Read; R/W = Read/Write; 'x' in a register name = 0-13; 'y' in a register name = 0-8.

**TABLE 3-6: SYSTEM BUS REGISTER MAP**

| Virtual Address<br>(BF8F_#) | Register Name | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |       | All Resets |       |       |
|-----------------------------|---------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|-------|------------|-------|-------|
|                             |               |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2  |            | 17/1  | 16/0  |
| 0510                        | SBFLAG        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —     | —          | —     | 0000  |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T3PGV | T2PGV      | T1PGV | T0PGV |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**TABLE 3-7: SYSTEM BUS TARGET 0 REGISTER MAP**

| Virtual Address<br>(BF8F_#) | Register Name | Bit Range | Bits        |       |       |       |             |       |      |           |      |          |      |      |        |            | All Resets |        |      |
|-----------------------------|---------------|-----------|-------------|-------|-------|-------|-------------|-------|------|-----------|------|----------|------|------|--------|------------|------------|--------|------|
|                             |               |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11       | 26/10 | 25/9 | 24/8      | 23/7 | 22/6     | 21/5 | 20/4 | 19/3   | 18/2       |            | 17/1   | 16/0 |
| 8020                        | SBT0ELOG1     | 31:16     | MULTI       | —     | —     | —     | CODE<3:0>   |       |      | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                             |               | 15:0      | INITID<7:0> |       |       |       | REGION<3:0> |       |      |           | —    | CMD<2:0> |      |      |        | 0000       |            |        |      |
| 8024                        | SBT0ELOG2     | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | GROUP<1:0> |            |        | 0000 |
| 8028                        | SBT0ECON      | 31:16     | —           | —     | —     | —     | —           | —     | ERRP | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
| 8030                        | SBT0ECLRS     | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | CLEAR  | 0000 |
| 8038                        | SBT0ECLRM     | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | CLEAR  | 0000 |
| 8040                        | SBT0REG0      | 31:16     | BASE<21:6>  |       |       |       |             |       |      |           |      |          |      |      |        |            | xxxx       |        |      |
|                             |               | 15:0      | BASE<5:0>   |       |       |       |             | PRI   | —    | SIZE<4:0> |      |          |      |      | —      | —          | —          | xxxx   |      |
| 8050                        | SBT0RD0       | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8058                        | SBT0WR0       | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8060                        | SBT0REG1      | 31:16     | BASE<21:6>  |       |       |       |             |       |      |           |      |          |      |      |        |            | xxxx       |        |      |
|                             |               | 15:0      | BASE<5:0>   |       |       |       |             | PRI   | —    | SIZE<4:0> |      |          |      |      | —      | —          | —          | xxxx   |      |
| 8070                        | SBT0RD1       | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8078                        | SBT0WR1       | 31:16     | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —    | —         | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note:** For reset values listed as 'xxxx', please refer to [Table 3-5](#) for the actual reset values.

TABLE 3-8: SYSTEM BUS TARGET 1 REGISTER MAP

| Virtual Address<br>(BF8F_#) | Register Name | Bit Range | Bits        |       |       |       |             |       |           |      |      |          |      |      |        |            | All Resets |        |
|-----------------------------|---------------|-----------|-------------|-------|-------|-------|-------------|-------|-----------|------|------|----------|------|------|--------|------------|------------|--------|
|                             |               |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11       | 26/10 | 25/9      | 24/8 | 23/7 | 22/6     | 21/5 | 20/4 | 19/3   | 18/2       |            | 17/1   |
| 8420                        | SBT1ELOG1     | 31:16     | MULTI       | —     | —     | —     | CODE<3:0>   |       |           |      | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                             |               | 15:0      | INITID<7:0> |       |       |       | REGION<3:0> |       |           |      | —    | CMD<2:0> |      |      |        | 0000       |            |        |
| 8424                        | SBT1ELOG2     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | GROUP<1:0> |            | 0000   |
| 8428                        | SBT1ECON      | 31:16     | —           | —     | —     | —     | —           | —     | ERRP      | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
| 8430                        | SBT1ECLRS     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | CLEAR      | 0000   |
| 8438                        | SBT1ECLRM     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | CLEAR      | 0000   |
| 8440                        | SBT1REG0      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                             |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 8450                        | SBT1RD0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8458                        | SBT1WR0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8480                        | SBT1REG2      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                             |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 8490                        | SBT1RD2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8498                        | SBT1WR2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 84A0                        | SBT1REG3      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                             |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 84B0                        | SBT1RD3       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 84B8                        | SBT1WR3       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 84C0                        | SBT1REG4      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                             |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 84D0                        | SBT1RD4       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 84D8                        | SBT1WR4       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                             |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note:** For reset values listed as 'xxxx', please refer to Table 3-5 for the actual reset values.

**TABLE 3-8: SYSTEM BUS TARGET 1 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF8F_#) | Register Name | Bit Range | Bits       |       |       |       |       |       |      |           |      |      |      |      |        |        | All Resets |        |
|-----------------------------|---------------|-----------|------------|-------|-------|-------|-------|-------|------|-----------|------|------|------|------|--------|--------|------------|--------|
|                             |               |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8      | 23/7 | 22/6 | 21/5 | 20/4 | 19/3   | 18/2   |            | 17/1   |
| 84E0                        | SBT1REG5      | 31:16     | BASE<21:6> |       |       |       |       |       |      |           |      |      |      |      |        |        | xxxx       |        |
|                             |               | 15:0      | BASE<5:0>  |       |       |       |       | PRI   | —    | SIZE<4:0> |      |      |      |      | —      | —      | —          | xxxx   |
| 84F0                        | SBT1RD5       | 31:16     | —          | —     | —     | —     | —     | —     | —    | —         | —    | —    | —    | —    | —      | —      | —          | xxxx   |
|                             |               | 15:0      | —          | —     | —     | —     | —     | —     | —    | —         | —    | —    | —    | —    | GROUP3 | GROUP2 | GROUP1     | GROUP0 |
| 84F8                        | SBT1WR5       | 31:16     | —          | —     | —     | —     | —     | —     | —    | —         | —    | —    | —    | —    | —      | —      | —          | xxxx   |
|                             |               | 15:0      | —          | —     | —     | —     | —     | —     | —    | —         | —    | —    | —    | —    | GROUP3 | GROUP2 | GROUP1     | GROUP0 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note:** For reset values listed as 'xxxx', please refer to [Table 3-5](#) for the actual reset values.



TABLE 3-9: SYSTEM BUS TARGET 2 REGISTER MAP

| Virtual Address (BF8F_#) | Register Name | Bit Range | Bits        |       |       |       |             |       |           |      |      |          |      |      |        |            | All Resets |        |
|--------------------------|---------------|-----------|-------------|-------|-------|-------|-------------|-------|-----------|------|------|----------|------|------|--------|------------|------------|--------|
|                          |               |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11       | 26/10 | 25/9      | 24/8 | 23/7 | 22/6     | 21/5 | 20/4 | 19/3   | 18/2       |            | 17/1   |
| 8820                     | SBT2ELOG1     | 31:16     | MULTI       | —     | —     | —     | CODE<3:0>   |       |           |      | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                          |               | 15:0      | INITID<7:0> |       |       |       | REGION<3:0> |       |           |      | —    | CMD<2:0> |      |      |        | 0000       |            |        |
| 8824                     | SBT2ELOG2     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | GROUP<1:0> |            | 0000   |
| 8828                     | SBT2ECON      | 31:16     | —           | —     | —     | —     | —           | —     | ERRP      | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
| 8830                     | SBT2ECLRS     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | CLEAR      | 0000   |
| 8838                     | SBT2ECLRM     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | 0000   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | CLEAR      | 0000   |
| 8840                     | SBT2REG0      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 8850                     | SBT2RD0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8858                     | SBT2WR0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8860                     | SBT2REG1      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 8870                     | SBT2RD1       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8878                     | SBT2WR1       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8880                     | SBT2REG2      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |
| 8890                     | SBT2RD2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |
| 8898                     | SBT2WR2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | xxxx   |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note:** For reset values listed as 'xxxx', please refer to Table 3-5 for the actual reset values.

TABLE 3-10: SYSTEM BUS TARGET 3 REGISTER MAP

| Virtual Address (BF8F_#) | Register Name | Bit Range | Bits        |       |       |       |             |       |           |      |      |          |      |      |        |            | All Resets |        |      |
|--------------------------|---------------|-----------|-------------|-------|-------|-------|-------------|-------|-----------|------|------|----------|------|------|--------|------------|------------|--------|------|
|                          |               |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11       | 26/10 | 25/9      | 24/8 | 23/7 | 22/6     | 21/5 | 20/4 | 19/3   | 18/2       |            | 17/1   | 16/0 |
| 8C20                     | SBT3ELOG1     | 31:16     | MULTI       | —     | —     | —     | CODE<3:0>   |       |           |      | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                          |               | 15:0      | INITID<7:0> |       |       |       | REGION<3:0> |       |           |      | —    | CMD<2:0> |      |      |        | 0000       |            |        |      |
| 8C24                     | SBT3ELOG2     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | GROUP<1:0> |            |        | 0000 |
| 8C28                     | SBT3ECON      | 31:16     | —           | —     | —     | —     | —           | —     | ERRP      | —    | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
| 8C30                     | SBT3ECLRS     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | CLEAR  | 0000 |
| 8C38                     | SBT3ECLRM     | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | 0000 |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | CLEAR  | 0000 |
| 8C40                     | SBT3REG0      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |      |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |      |
| 8C50                     | SBT3RD0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8C58                     | SBT3WR0       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8C60                     | SBT3REG1      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |      |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |      |
| 8C70                     | SBT3RD1       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8C78                     | SBT3WR1       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8C80                     | SBT3REG2      | 31:16     | BASE<21:6>  |       |       |       |             |       |           |      |      |          |      |      |        |            | xxxx       |        |      |
|                          |               | 15:0      | BASE<5:0>   |       |       |       | PRI         | —     | SIZE<4:0> |      |      |          | —    | —    | —      | —          | xxxx       |        |      |
| 8C90                     | SBT3RD2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |
| 8C98                     | SBT3WR2       | 31:16     | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | —      | —          | —          | —      | xxxx |
|                          |               | 15:0      | —           | —     | —     | —     | —           | —     | —         | —    | —    | —        | —    | —    | GROUP3 | GROUP2     | GROUP1     | GROUP0 | xxxx |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note:** For reset values listed as 'xxxx', please refer to Table 3-5 for the actual reset values.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 3-1: SBFLAG: SYSTEM BUS STATUS FLAG REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | R-0            | R-0            | R-0           | R-0           |
|           | —              | —              | —              | —              | T3PGV          | T2PGV          | T1PGV         | T0PGV         |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

bit 31-4 **Unimplemented:** Read as '0'

bit 3-0 **T3PGV:T0PGV:** Target Permission Group Violation Status bits

Refer to [Table 3-5](#) for the list of available targets and their descriptions.

1 = Target is reporting a Permission Group (PG) violation

0 = Target is not reporting a PG violation

**Note:** All errors are cleared at the source (i.e., SBTxELOG1, SBTxELOG2, SBTxECLRS, or SBTxECLRM registers).

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 3-2: SBTxELOG1: SYSTEM BUS TARGET 'x' ERROR LOG REGISTER 1 ('x' = 0-3)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0, C       | U-0            | U-0            | U-0            | R/W-0, C       | R/W-0, C       | R/W-0, C      | R/W-0, C      |
|           | MULTI          | —              | —              | —              | CODE<3:0>      |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | INITID<7:0>    |                |                |                |                |                |               |               |
| 7:0       | R-0            | R-0            | R-0            | R-0            | U-0            | R-0            | R-0           | R-0           |
|           | REGION<3:0>    |                |                |                | —              | CMD<2:0>       |               |               |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | C = Clearable bit                  |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |

bit 31 **MULTI:** Multiple Permission Violations Status bit

This bit is cleared by writing a '1'.

- 1 = Multiple errors have been detected
- 0 = No multiple errors have been detected

bit 30-28 **Unimplemented:** Read as '0'

bit 27-24 **CODE<3:0>:** Error Code bits

Indicates the type of error that was detected. These bits are cleared by writing a '1'.

- 1111 = Reserved
- 1101 = Reserved
- .
- .
- .
- 0011 = Permission violation
- 0010 = Reserved
- 0001 = Reserved
- 0000 = No error

bit 23-16 **Unimplemented:** Read as '0'

bit 15-8 **INITID<7:0>:** Initiator ID of Requester bits

- 11111111 = Reserved
- .
- .
- .
- 00001111 = Reserved
- 00001110 = Reserved
- 00001101 = Reserved
- 00001100 = Reserved
- 00001011 = Reserved
- 00001010 = CAN1
- 00001001 = Reserved
- 00001000 = Reserved
- 00000111 = ADC0-ADC5, ADC7
- 00000110 = Reserved
- 00000101 = Flash Controller
- 00000100 = DMA Read
- 00000011 = DMA Read
- 00000010 = CPU (CPUPRI (CFGCON<24>) = 1)
- 00000001 = CPU (CPUPRI (CFGCON<25>) = 0)
- 00000000 = Reserved

**Note:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 3-2: SBTxELOG1: SYSTEM BUS TARGET 'x' ERROR LOG REGISTER 1 ( 'x' = 0-3) (CONTINUED)

- bit 7-4 **REGION<3:0>**: Requested Region Number bits  
1111 - 0000 = Target's region that reported a permission group violation
- bit 3 **Unimplemented**: Read as '0'
- bit 2-0 **CMD<2:0>**: Transaction Command of the Requester bits  
111 = Reserved  
110 = Reserved  
101 = Write (a non-posted write)  
100 = Reserved  
011 = Read (a locked read caused by a Read-Modify-Write transaction)  
010 = Read  
001 = Write  
000 = Idle

**Note:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 3-5: SBTxECLRS: SYSTEM BUS TARGET 'x' SINGLE ERROR CLEAR REGISTER ('x' = 0-3)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R-0           |
|           | —              | —              | —              | —              | —              | —              | —             | CLEAR         |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

bit 31-1 **Unimplemented:** Read as '0'

bit 0 **CLEAR:** Clear Single Error on Read bit

A single error as reported through SBTxELOG1 and SBTxELOG2 is cleared by a read of this register.

**Note:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

## REGISTER 3-6: SBTxECLRM: SYSTEM BUS TARGET 'x' MULTIPLE ERROR CLEAR REGISTER ('x' = 0-3)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R-0           |
|           | —              | —              | —              | —              | —              | —              | —             | CLEAR         |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

bit 31-1 **Unimplemented:** Read as '0'

bit 0 **CLEAR:** Clear Multiple Errors on Read bit

Multiple errors as reported through SBTxELOG1 and SBTxELOG2 is cleared by a read of this register.

**Note:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 3-7: SBTxREGy: SYSTEM BUS TARGET 'x' REGION 'y' REGISTER (‘x’ = 0-3; ‘y’ = 0-2)

| Bit Range   | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24       | R/W0           | R/W-0          | R/W0           | R/W-0          | R/W0           | R/W-0          | R/W0          | R/W-0         |
| BASE<21:14> |                |                |                |                |                |                |               |               |
| 23:16       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| BASE<13:6>  |                |                |                |                |                |                |               |               |
| 15:8        | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R-0           | U-0           |
| BASE<5:0>   |                |                |                |                |                |                | PRI           | —             |
| 7:0         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | U-0            | U-0           | U-0           |
| SIZE<4:0>   |                |                |                |                |                | —              | —             | —             |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as ‘0’

-n = Value at POR

‘1’ = Bit is set

‘0’ = Bit is cleared

bit 31-10 **BASE<21:0>**: Region Base Address bits

bit 9 **PRI**: Region Priority Level bit

1 = Level 2

0 = Level 1

bit 8 **Unimplemented**: Read as ‘0’

bit 7-3 **SIZE<4:0>**: Region Size bits

Permissions for a region are only active if the SIZE is non-zero.

11111 = Region size =  $2^{(SIZE - 1)} \times 1024$  (bytes)

•

•

•

00001 = Region size =  $2^{(SIZE - 1)} \times 1024$  (bytes)

00000 = Region is not present

bit 2-0 **Unimplemented**: Read as ‘0’

**Note 1:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

**Note 2:** For some target regions, certain bits in this register are read-only with preset values. See [Table 3-5](#) for more information.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 3-8: SBTxRDy: SYSTEM BUS TARGET 'x' REGION 'y' READ PERMISSIONS REGISTER ('x' = 0-3; 'y' = 0-2)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | R-1            | R-1            | R-1           | R-1           |
|           | —              | —              | —              | —              | GROUP3         | GROUP2         | GROUP1        | GROUP0        |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared

- bit 31-4 **Unimplemented:** Read as '0'
- bit 3 **GROUP3:** Group 3 Read Permissions bits
  - 1 = Privilege Group 3 has read permission
  - 0 = Privilege Group 3 does not have read permission
- bit 2 **GROUP2:** Group 2 Read Permissions bits
  - 1 = Privilege Group 2 has read permission
  - 0 = Privilege Group 2 does not have read permission
- bit 1 **GROUP1:** Group 1 Read Permissions bits
  - 1 = Privilege Group 1 has read permission
  - 0 = Privilege Group 1 does not have read permission
- bit 0 **GROUP0:** Group 0 Read Permissions bits
  - 1 = Privilege Group 0 has read permission
  - 0 = Privilege Group 0 does not have read permission

**Note 1:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

**Note 2:** For some target regions, certain bits in this register are read-only with preset values. See [Table 3-5](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 3-9: SBTxWRy: SYSTEM BUS TARGET 'x' REGION 'y' WRITE PERMISSIONS REGISTER ('x' = 0-3; 'y' = 0-2)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | R/W-1          | R/W-1          | R/W-1         | R/W-1         |
|           | —              | —              | —              | —              | GROUP3         | GROUP2         | GROUP1        | GROUP0        |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

bit 31-4 **Unimplemented:** Read as '0'

bit 3 **GROUP3:** Group 3 Write Permissions bits  
 1 = Privilege Group 3 has write permission  
 0 = Privilege Group 3 does not have write permission

bit 2 **GROUP2:** Group 2 Write Permissions bits  
 1 = Privilege Group 2 has write permission  
 0 = Privilege Group 2 does not have write permission

bit 1 **GROUP1:** Group 1 Write Permissions bits  
 1 = Privilege Group 1 has write permission  
 0 = Privilege Group 1 does not have write permission

bit 0 **GROUP0:** Group 0 Write Permissions bits  
 1 = Privilege Group 0 has write permission  
 0 = Privilege Group 0 does not have write permission

**Note 1:** Refer to [Table 3-5](#) for the list of available targets and their descriptions.

**Note 2:** For some target regions, certain bits in this register are read-only with preset values. See [Table 3-5](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

## 4.0 FLASH PROGRAM MEMORY

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 52. “Flash Program Memory”** (DS60001193), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

PIC32MK GPG/MCJ with CAN FD Family of devices contain an internal Flash program memory for executing user code, which includes the following features:

- Write protection for program and boot Flash
- ECC support

There are three methods by which the user can program this memory:

- Run-Time Self-Programming (RTSP)
- EJTAG Programming
- In-Circuit Serial Programming (ICSP)

RTSP is performed by software executing from either Flash or RAM memory. For information about RTSP techniques, refer to **Section 52. “Flash Program Memory”** (DS60001193) in the *“PIC32 Family Reference Manual”*.

EJTAG is performed using the EJTAG port of the device and an EJTAG capable programmer.

ICSP is performed using a serial data connection to the device and allows much faster programming times than RTSP.

The EJTAG and ICSP methods are described in the *“PIC32 Flash Programming Specification”* (DS60001145), which is available for download from the Microchip website.

**Note:** In PIC32MK GPG/MCJ with CAN FD Family of devices, the Flash page size is 1024 Instruction Words and the row size is 128 Instruction Words.

4.1 Flash Control Registers

TABLE 4-1: FLASH CONTROLLER REGISTER MAP

| Virtual Address (BF80_#) | Register Name          | Bit Range | Bits             |       |        |        |       |       |             |       |               |            |      |       |       |            |       | All Resets |      |
|--------------------------|------------------------|-----------|------------------|-------|--------|--------|-------|-------|-------------|-------|---------------|------------|------|-------|-------|------------|-------|------------|------|
|                          |                        |           | 31/15            | 30/14 | 29/13  | 28/12  | 27/11 | 26/10 | 25/9        | 24/8  | 23/7          | 22/6       | 21/5 | 20/4  | 19/3  | 18/2       | 17/1  |            | 16/0 |
| 0A00                     | NVMCON <sup>(1)</sup>  | 31:16     | —                | —     | —      | —      | —     | —     | —           | —     | —             | —          | —    | —     | —     | —          | —     | —          | 0000 |
|                          |                        | 15:0      | WR               | WREN  | WRERR  | LVDERR | —     | —     | —           | —     | —             | —          | —    | —     | —     | NVMOP<3:0> |       |            | 0000 |
| 0A10                     | NVMKEY                 | 31:16     | NVMKEY<31:0>     |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A20                     | NVMADDR <sup>(1)</sup> | 31:16     | NVMADDR<31:0>    |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A30                     | NVMDATA0               | 31:16     | NVMDATA0<31:0>   |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A40                     | NVMDATA1               | 31:16     | NVMDATA1<31:0>   |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A50                     | NVMDATA2               | 31:16     | NVMDATA2<31:0>   |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A60                     | NVMDATA3               | 31:16     | NVMDATA3<31:0>   |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A70                     | NVMSRC ADDR            | 31:16     | NVMSRCADDR<31:0> |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
|                          |                        | 15:0      |                  |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A80                     | NVMPWP <sup>(1)</sup>  | 31:16     | PWPUNLOCK        | —     | —      | —      | —     | —     | —           | —     | —             | PWP<23:16> |      |       |       |            |       | 8000       |      |
|                          |                        | 15:0      | PWP<15:0>        |       |        |        |       |       |             |       |               |            |      |       |       |            |       | 0000       |      |
| 0A90                     | NVMBWP <sup>(1)</sup>  | 31:16     | —                | —     | —      | —      | —     | —     | —           | —     | —             | —          | —    | —     | —     | —          | —     | —          | 0000 |
|                          |                        | 15:0      | LBWPUNLOCK       | —     | —      | LBWP4  | LBWP3 | LBWP2 | LBWP1       | LBWP0 | UBWPUNLOCK    | —          | —    | UBWP4 | UBWP3 | UBWP2      | UBWP1 | UBWP0      | 9FDF |
| 0AA0                     | NVMCON2 <sup>(1)</sup> | 31:16     | ERSCNT<3:0>      |       |        | —      | —     | —     | —           | —     | —             | —          | —    | —     | —     | —          | —     | —          | 001F |
|                          |                        | 15:0      | —                | —     | CREAD1 | VREAD1 | —     | —     | ERETRY<1:0> |       | SWAPLOCK<1:0> |            | —    | —     | —     | —          | —     | —          | 0000 |

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: This register has corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See 10.2 "CLR, SET, and INV Registers" for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 4-1: NVMCON: PROGRAMMING CONTROL REGISTER**

| Bit Range  | Bit 31/23/15/7                 | Bit 30/22/14/6               | Bit 29/21/13/5                      | Bit 28/20/12/4                       | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|------------|--------------------------------|------------------------------|-------------------------------------|--------------------------------------|----------------|----------------|---------------|---------------|
| 31:24      | U-0<br>—                       | U-0<br>—                     | U-0<br>—                            | U-0<br>—                             | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 23:16      | U-0<br>—                       | U-0<br>—                     | U-0<br>—                            | U-0<br>—                             | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 15:8       | R/W-0, HC<br>WR <sup>(1)</sup> | R/W-0<br>WREN <sup>(1)</sup> | R-0, HS, HC<br>WRERR <sup>(1)</sup> | R-0, HS, HC<br>LVDERR <sup>(1)</sup> | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 7:0        | U-0<br>—                       | U-0<br>—                     | U-0<br>—                            | U-0<br>—                             | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMOP<3:0> |                                |                              |                                     |                                      |                |                |               |               |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **WR:** Write Control bit<sup>(1)</sup>

This bit cannot be cleared and can be set only when WREN = 1 and the unlock sequence has been performed.

- 1 = Initiate a Flash operation
- 0 = Flash operation is complete or inactive

bit 14 **WREN:** Write Enable bit<sup>(1)</sup>

- 1 = Enable writes to the WR bit and disables writes to the NVMOP<3:0> bits
- 0 = Disable writes to WR bit and enables writes to the NVMOP<3:0> bits

bit 13 **WRERR:** Write Error bit<sup>(1)</sup>

This bit can be cleared only by setting the NVMOP<3:0> bits = 0000 and initiating a Flash operation.

- 1 = Program or erase sequence did not complete successfully
- 0 = Program or erase sequence completed normally

bit 12 **LVDERR:** Low-Voltage Detect Error bit<sup>(1)</sup>

This bit can be cleared only by setting the NVMOP<3:0> bits = 0000 and initiating a Flash operation.

- 1 = Low-voltage detected (possible data corruption, if WRERR is set)
- 0 = Voltage level is acceptable for programming

bit 11-4 **Unimplemented:** Read as '0'

**Note 1:** These bits are only reset by a Power-on Reset (POR) and are not affected by other reset sources.

- 2: This bit can only be modified when the WREN bit = 0, the NVMKEY unlock sequence is satisfied, and the SWAPLOCK<1:0> bits (NVMCON2<7:6>) are cleared to '0'.
- 3: This operation results in a "No Operation" (NOP) when the Dynamic Flash ECC Configuration bits = 00 (FECCCON<1:0> (DVCFG0<9:8>)), which enables ECC at all times. For all other FECCCON<1:0> bit settings, this command will execute, but will not write the ECC bits for the word and can cause DED errors if dynamic Flash ECC is enabled (FECCCON<1:0> = 01). Refer to **Section 52. "Flash Program Memory"** (DS60001193) for information regarding ECC and Flash programming.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 4-1: NVMCON: PROGRAMMING CONTROL REGISTER (CONTINUED)

bit 3-0 **NVMOP<3:0>**: NVM Operation bits

These bits are only writable when WREN = 0.

1111 = Reserved

•  
•  
•

1000 = Reserved

0111 = Program erase operation: erase all of program Flash memory (all pages must be unprotected, PWP<23:0> = 0x000000)

0110 = Reserved

0101 = Program Flash memory erase operation: erases all of Program Flash (all pages in that region must be unprotected)

0100 = Page erase operation: erases page selected by NVMADDR, if it is not write-protected

0011 = Row program operation: programs row selected by NVMADDR, if it is not write-protected

0010 = Quad Word (128-bit) program operation: programs the 128-bit Flash word selected by NVMADDR, if it is not write-protected

0001 = Word program operation: programs word selected by NVMADDR, if it is not write-protected<sup>(4)</sup>

0000 = No operation

- Note 1:** These bits are only reset by a Power-on Reset (POR) and are not affected by other reset sources.
- 2:** This bit can only be modified when the WREN bit = 0, the NVMKEY unlock sequence is satisfied, and the SWAPLOCK<1:0> bits (NVMCON2<7:6>) are cleared to '0'.
- 3:** This operation results in a "No Operation" (NOF) when the Dynamic Flash ECC Configuration bits = 00 (FECCCON<1:0> (DVCFG0<9:8>)), which enables ECC at all times. For all other FECCCON<1:0> bit settings, this command will execute, but will not write the ECC bits for the word and can cause DED errors if dynamic Flash ECC is enabled (FECCCON<1:0> = 01). Refer to **Section 52. "Flash Program Memory"** (DS60001193) for information regarding ECC and Flash programming.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 4-2: NVMKEY: PROGRAMMING UNLOCK REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | W-0            | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
|           | NVMKEY<31:24>  |                |                |                |                |                |               |               |
| 23:16     | W-0            | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
|           | NVMKEY<23:16>  |                |                |                |                |                |               |               |
| 15:8      | W-0            | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
|           | NVMKEY<15:8>   |                |                |                |                |                |               |               |
| 7:0       | W-0            | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
|           | NVMKEY<7:0>    |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **NVMKEY<31:0>**: Unlock Register bits  
 These bits are write-only, and read as '0' on any read

**Note:** This register is used as part of the unlock sequence to prevent inadvertent writes to the PFM.

## REGISTER 4-3: NVMADDR: FLASH ADDRESS REGISTER

| Bit Range | Bit 31/23/15/7                | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-------------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0                         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NVMADDR<31:24> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 23:16     | R/W-0                         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NVMADDR<23:16> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 15:8      | R/W-0                         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NVMADDR<15:8> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 7:0       | R/W-0                         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NVMADDR<7:0> <sup>(1)</sup>   |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **NVMADDR<31:0>**: Flash Address bits<sup>(1)</sup>

| NVMOP<3:0> Selection | Flash Address Bits (NVMADDR<31:0>)   |
|----------------------|--|
| Page Erase           | Address identifies the page to erase (NVMADDR<11:0> are ignored).                      |
| Row Program          | Address identifies the row to program (NVMADDR<8:0> are ignored).                      |
| Word Program         | Address identifies the word to program (NVMADDR<1:0> are ignored).                     |
| Quad Word Program    | Address identifies the quad word (128-bit) to program (NVMADDR<3:0> bits are ignored). |

**Note 1:** For all other NVMOP<3:0> bit settings, the Flash address is ignored. See the NVMCON register ([Register 4-1](#)) for additional information on these bits.

**Note:** The bits in this register are only reset by a POR and are not affected by other reset sources.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 4-4: NVMDATAx: FLASH DATA REGISTER (x = 0-3)**

| Bit Range      | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMDATA<31:24> |                |                |                |                |                |                |               |               |
| 23:16          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMDATA<23:16> |                |                |                |                |                |                |               |               |
| 15:8           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMDATA<15:8>  |                |                |                |                |                |                |               |               |
| 7:0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMDATA<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **NVMDATA<31:0>**: Flash Data bits  
 Word Program: Writes NVMDATA0 to the target Flash address defined in NVMADDR  
 Quad Word Program: Writes NVMDATA3:NVMDATA2:NVMDATA1:NVMDATA0 to the target Flash address defined in NVMADDR. NVMDATA0 contains the Least Significant Instruction Word.

**Note:** The bits in this register are only reset by a POR and are not affected by other reset sources.

**REGISTER 4-5: NVMSRCADDR: SOURCE DATA ADDRESS REGISTER**

| Bit Range         | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-------------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24             | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMSRCADDR<31:24> |                |                |                |                |                |                |               |               |
| 23:16             | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMSRCADDR<23:16> |                |                |                |                |                |                |               |               |
| 15:8              | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMSRCADDR<15:8>  |                |                |                |                |                |                |               |               |
| 7:0               | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NVMSRCADDR<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **NVMSRCADDR<31:0>**: Source Data Address bits  
 The system physical address of the data to be programmed into the Flash when the NVMOP<3:0> bits (NVMCON<3:0>) are set to perform row programming.

**Note:** The bits in this register are only reset by a POR and are not affected by other reset sources.





# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 4-7: NVMBWP: FLASH BOOT (PAGE) WRITE-PROTECT REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4      | Bit 27/19/11/3      | Bit 26/18/10/2      | Bit 25/17/9/1       | Bit 24/16/8/0       |
|-----------|----------------|----------------|----------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| 31:24     | U-0            | U-0            | U-0            | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 |
|           | —              | —              | —              | —                   | —                   | —                   | —                   | —                   |
| 23:16     | U-0            | U-0            | U-0            | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 |
|           | —              | —              | —              | —                   | —                   | —                   | —                   | —                   |
| 15:8      | R/W-1          | U-0            | U-0            | R/W-1               | R/W-1               | R/W-1               | R/W-1               | R/W-1               |
|           | BWPULOCK       | —              | —              | BWP4 <sup>(1)</sup> | BWP3 <sup>(1)</sup> | BWP2 <sup>(1)</sup> | BWP1 <sup>(1)</sup> | BWP0 <sup>(1)</sup> |
| 7:0       | r-1            | r-1            | U-0            | r-1                 | r-1                 | r-1                 | r-1                 | r-1                 |
|           | —              | —              | —              | —                   | —                   | —                   | —                   | —                   |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved                       |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **BWPULOCK:** Lower Boot Alias Write-protect Unlock bit

1 = LBWPx bits are not locked and can be modified

0 = LBWPx bits are locked and cannot be modified

This bit is only clearable and cannot be set except by any reset.

bit 14-13 **Unimplemented:** Read as '0'

bit 12 **BWP4:** Boot Alias Page 4 Write-protect bit<sup>(1)</sup>

1 = Write protection for physical address 0x01FC10000 through 0x1FC13FFF enabled

0 = Write protection for physical address 0x01FC10000 through 0x1FC13FFF disabled

bit 11 **BWP3:** Boot Alias Page 3 Write-protect bit<sup>(1)</sup>

1 = Write protection for physical address 0x01FC0C000 through 0x1FC0FFFF enabled

0 = Write protection for physical address 0x01FC0C000 through 0x1FC0FFFF disabled

bit 10 **BWP2:** Boot Alias Page 2 Write-protect bit<sup>(1)</sup>

1 = Write protection for physical address 0x01FC08000 through 0x1FC0BFFF enabled

0 = Write protection for physical address 0x01FC08000 through 0x1FC0BFFF disabled

bit 9 **BWP1:** Boot Alias Page 1 Write-protect bit<sup>(1)</sup>

1 = Write protection for physical address 0x01FC04000 through 0x1FC07FFF enabled

0 = Write protection for physical address 0x01FC04000 through 0x1FC07FFF disabled

bit 8 **BWP0:** Boot Alias Page 0 Write-protect bit<sup>(1)</sup>

1 = Write protection for physical address 0x01FC00000 through 0x1FC03FFF enabled

0 = Write protection for physical address 0x01FC00000 through 0x1FC03FFF disabled

bit 7 **Reserved**

bit 6 **Reserved:** This bit is reserved for use by development tools

bit 5 **Unimplemented:** Read as '0'

bit 4-0 **Unimplemented:** Read as '0'

**Note:** The bits in this register are only writable when the NVMKEY unlock sequence is followed.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 4-8: NVMCON2: FLASH PROGRAMMING CONTROL REGISTER 2

| Bit Range | Bit 31/23/15/7   | Bit 30/22/14/6 | Bit 29/21/13/5        | Bit 28/20/12/4        | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|--|----------------|-----------------------|-----------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0  | R/W-0          | R/W-0                 | R/W-0                 | U-0            | U-0            | U-0           | U-0           |
|           | ERSCNT<3:0>  |                |                       |                       | —              | —              | —             | —             |
| 23:16     | U-0  | U-0            | U-0                   | r-1                   | r-1            | r-1            | r-1           | r-1           |
|           | (Reserved, users must always write '1' to these locations) |                |                       |                       |                |                |               |               |
| 15:8      | r-0  | U-0            | R/W-0                 | R/W-0                 | U-0            | U-0            | R/W-0         | R/W-0         |
|           | —  | —              | CREAD1 <sup>(1)</sup> | VREAD1 <sup>(1)</sup> | —              | —              | ERETRY<1:0>   |               |
| 7:0       | U-0  | U-0            | U-0                   | U-0                   | U-0            | U-0            | U-0           | U-0           |
|           | —  | —              | —                     | —                     | —              | —              | —             | —             |

**Legend:** r = Reserved

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-28 **ERSCNT<1:0>**: Erase Retry State Count bits

These bits can be used by software to track the erase retry state count in the event of a Master Clear or BOR. These bits are purely for software tracking purpose and are not used by hardware in any way.

bit 27-21 **Unimplemented**: Read as '0'

bit 20-16 **Reserved**: Users must always write '1' to these bits

bit 15-14 **Unimplemented**: Read as '0'

bit 13 **CREAD1**: Compare Read of Logic 1 bit<sup>(1)</sup>

1 = Compare Read is enabled (only if VERIFYREAD1 = 1)

0 = Compare Read is disabled

Compare Read '1' causes all bits in a Flash Word to be evaluated during the read. If all bits are '1', the lowest Word in the Flash Word evaluates to 0x00000001, all other words are 0x00010000. If any bit is '0', the read evaluates to 0x00000000 for all Words in the Flash Word.

bit 12 **VREAD1**: Verify Read of Logic 1 Control bit<sup>(1)</sup>

1 = Selects Erase Retry Procedure with Verify Read

0 = Selects Single Erase w/o Verify Read

bit 11-10 **Unimplemented**: Read as '0'

bit 9-8 **ERETRY<1:0>**: Erase Retry Control bits

11 = Erase strength for last retry cycle

10 = Erase strength for third retry cycle

01 = Erase strength for second retry cycle

00 = Erase strength for first retry cycle

The user application must start with '00' (first retry cycle) and move on to higher strength if the programming does not complete.

This bit is used only when VREAD1 = 1 and when VREAD1 = 1.

bit 7-0 **Unimplemented**: Read as '0'

**Note 1:** This bit can only be modified when the WREN bit = 0, and the NVMKEY unlock sequence is satisfied.

# PIC32MK GPG/MCJ with CAN FD Family

## 5.0 RESETS

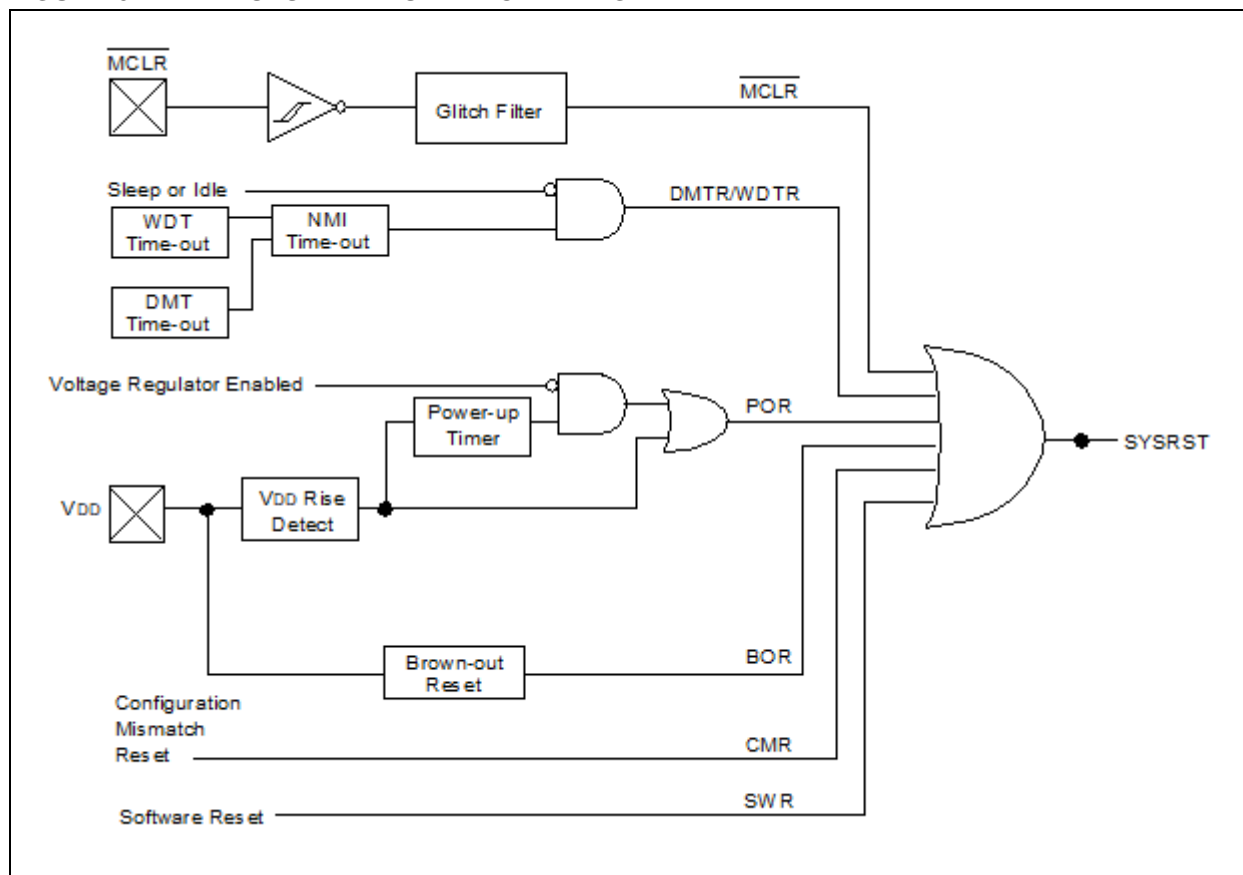
**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 7. “Resets”** (DS60001118), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Reset module combines all Reset sources and controls the device Master Reset signal, SYSRST. The device Reset sources are as follows:

- Power-on Reset (POR)
- Master Clear Reset pin ( $\overline{\text{MCLR}}$ )
- Software Reset (SWR)
- Watchdog Timer Reset (WDTR)
- Brown-out Reset (BOR)
- Configuration Mismatch Reset (CMR)
- Deadman Timer Reset (DMTR)

A simplified block diagram of the Reset module is illustrated in [Figure 5-1](#).

**FIGURE 5-1: SYSTEM RESET BLOCK DIAGRAM**



5.1 Reset Control Registers

TABLE 5-1: RESETS REGISTER MAP

| Virtual Address<br>(BF80_#) | Register Name | Bit Range | Bits      |       |       |       |         |          |      |      |       |              |              |      |       |      |      | All Resets |       |      |
|-----------------------------|---------------|-----------|-----------|-------|-------|-------|---------|----------|------|------|-------|--------------|--------------|------|-------|------|------|------------|-------|------|
|                             |               |           | 31/15     | 30/14 | 29/13 | 28/12 | 27/11   | 26/10    | 25/9 | 24/8 | 23/7  | 22/6         | 21/5         | 20/4 | 19/3  | 18/2 | 17/1 |            | 16/0  |      |
| 1240                        | RCON          | 31:16     | —         | —     | —     | —     | BCFGERR | BCFGFAIL | —    | —    | —     | —            | —            | —    | —     | —    | —    | —          | —     | C000 |
|                             |               | 15:0      | —         | —     | —     | —     | —       | —        | CMR  | —    | EXTR  | SWR          | DMTO         | WDTO | SLEEP | IDLE | BOR  | POR        | 0003  |      |
| 1250                        | RSWRST        | 31:16     | —         | —     | —     | —     | —       | —        | —    | —    | —     | —            | —            | —    | —     | —    | —    | —          | —     | 0000 |
|                             |               | 15:0      | —         | —     | —     | —     | —       | —        | —    | —    | —     | —            | —            | —    | —     | —    | —    | —          | SWRST | 0000 |
| 1260                        | RNMICON       | 31:16     | —         | —     | —     | —     | —       | —        | DMTO | WDTO | SWNMI | —            | —            | —    | GNMI  | HLVD | CF   | —          | —     | 0000 |
|                             |               | 15:0      | NMI<15:0> |       |       |       |         |          |      |      |       |              |              |      |       |      |      | 0000       |       |      |
| 1270                        | PWRCON        | 31:16     | —         | —     | —     | —     | —       | —        | —    | —    | —     | —            | —            | —    | —     | —    | —    | —          | —     | 0000 |
|                             |               | 15:0      | —         | —     | —     | —     | —       | —        | —    | —    | —     | VREGRUN<1:0> | VREGSLP<1:0> | —    | —     | —    | —    | —          | VREGS | 0000 |

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 5-1: RCON: RESET CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1      | Bit 24/16/8/0      |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|--------------------|--------------------|
| 31:24     | U-1, HS        | U-1, HS        | U-0            | U-0            | R/W-0, HC      | R/W-0, HC      | U-0                | U-0                |
|           | —              | —              | —              | —              | BCFGERR        | BCFGFAIL       | —                  | —                  |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-1                | U-x                |
|           | —              | —              | —              | —              | —              | —              | —                  | —                  |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0, HS          | U-0                |
|           | —              | —              | —              | —              | —              | —              | CMR                | —                  |
| 7:0       | R/W-0, HS      | R/W-0, HS      | R/W-0, HS      | R/W-0, HS      | R/W-0, HS      | R/W-0, HS      | R/W-1, HS          | R/W-1, HS          |
|           | EXTR           | SWR            | DMTO           | WDTO           | SLEEP          | IDLE           | BOR <sup>(2)</sup> | POR <sup>(2)</sup> |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31:30 **Reserved:** Read as '1' on power up.

bit 29-28 **Unimplemented:** Read as '0'

bit 27 **BCFGERR:** Primary Configuration Registers Error Flag bit

1 = An error occurred during a read of the primary configuration registers

0 = No error occurred during a read of the primary configuration registers

bit 26 **BCFGFAIL:** Primary/Secondary Configuration Registers Error Flag bit

1 = An error occurred during a read of the primary and alternate configuration registers

0 = No error occurred during a read of the primary and alternate configuration registers

bit 25-10 **Unimplemented:** Read as '0'

bit 9 **CMR:** Configuration Mismatch Reset Flag bit

1 = A Configuration Mismatch Reset has occurred

0 = A Configuration Mismatch Reset has not occurred

bit 8 **Unimplemented:** Read as '0'

bit 7 **EXTR:** External Reset ( $\overline{\text{MCLR}}$ ) Pin Flag bit

1 = Master Clear (pin) Reset has occurred

0 = Master Clear (pin) Reset has not occurred

bit 6 **SWR:** Software Reset Flag bit

1 = Software Reset was executed

0 = Software Reset was not executed

bit 5 **DMTO:** Deadman Timer Time-out Flag bit

1 = A DMT time-out has occurred

0 = A DMT time-out has not occurred

bit 4 **WDTO:** Watchdog Timer Time-out Flag bit

1 = WDT Time-out has occurred

0 = WDT Time-out has not occurred

bit 3 **SLEEP:** Wake From Sleep Flag bit

1 = Device was in Sleep mode

0 = Device was not in Sleep mode

bit 2 **IDLE:** Wake From Idle Flag bit

1 = Device was in Idle mode

0 = Device was not in Idle mode

**Note 1:** User software must clear this bit to view the next detection.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 5-1: RCON: RESET CONTROL REGISTER

- bit 1     **BOR:** Brown-out Reset Flag bit<sup>(1)</sup>  
          1 = Brown-out Reset has occurred  
          0 = Brown-out Reset has not occurred
- bit 0     **POR:** Power-on Reset Flag bit<sup>(1)</sup>  
          1 = Power-on Reset has occurred  
          0 = Power-on Reset has not occurred

**Note 1:** User software must clear this bit to view the next detection.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 5-2: RSWRST: SOFTWARE RESET REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0          |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|------------------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0                    |
|           | —              | —              | —              | —              | —              | —              | —             | —                      |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0                    |
|           | —              | —              | —              | —              | —              | —              | —             | —                      |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0                    |
|           | —              | —              | —              | —              | —              | —              | —             | —                      |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | W-0, HC                |
|           | —              | —              | —              | —              | —              | —              | —             | SWRST <sup>(1,2)</sup> |

|                   |  |
|-------------------|--|
| <b>Legend:</b>    | HC = Hardware Cleared  |
| R = Readable bit  | W = Writable bit      U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown |

bit 31-1 **Unimplemented:** Read as '0'

bit 0 **SWRST:** Software Reset Trigger bit<sup>(1,2)</sup>  
 1 = Enable software Reset event  
 0 = No effect

- Note 1:** The system unlock sequence must be performed before the SWRST bit can be written. Refer to **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the "PIC32 Family Reference Manual" for details.
- 2:** Once this bit is set, any read of the RSWRST register will cause a Reset to occur.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 5-3: RNMICON: NON-MASKABLE INTERRUPT (NMI) CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | —              | —              | DMTO          | WDTO          |
| 23:16     | R/W-0          | U-0            | U-0            | U-0            | R/W-0          | R/W-0, HS      | R/W-0, HS, HC | R/W-0         |
|           | SWNMI          | —              | —              | —              | GNMI           | HLVD           | CF            | U-0           |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | —             |
|           | NMICNT<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NMICNT<7:0>    |                |                |                |                |                |               |               |

|                   |                     |  |
|-------------------|---------------------|--|
| <b>Legend:</b>    | HC = Hardware Clear | HS = Hardware Set                            |
| R = Readable bit  | W = Writable bit    | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set    | '0' = Bit is cleared      x = Bit is unknown |

bit 31-26 **Unimplemented:** Read as '0'

bit 25 **DMTO:** Deadman Timer Time-out Flag bit

1 = DMT time-out has occurred and caused a NMI

0 = DMT time-out has not occurred

Setting this bit will cause a DMT NMI event, and NMICNT will begin counting. Clearing this bit before the NMICNT SYSCLK" cycle counter has expired will negate a normal DMT CPU reset.

bit 24 **WDTO:** Watchdog Timer Time-Out Flag bit

1 = WDT time-out has occurred and caused a NMI

0 = WDT time-out has not occurred

Setting this bit will cause a WDT NMI event, and NMICNT will begin counting. Clearing this bit before the NMICNT SYSCLK" cycle counter has expired will negate a normal WDT CPU reset.

bit 23 **SWNMI:** Software NMI Trigger.

1 = An NMI will be generated

0 = An NMI will not be generated

bit 22-20 **Unimplemented:** Read as '0'

bit 19 **GNMI:** General NMI bit

1 = A general NMI event has been detected or a user-initiated NMI event has occurred

0 = A general NMI event has not been detected

Setting GNMI to a '1' causes a user-initiated NMI event. This bit is also set by writing 0x4E to the NMIKEY<7:0> (INTCON<31:24>) bits.

bit 18 **HLVD:** High/Low-Voltage Detect Status

1 = Indicates HLVD Event interrupt is active.

0 = Indicates HLVD Event interrupt is not active.

**Note 1:** When a Watchdog Timer NMI event (when not in Sleep mode) or a Deadman Timer NMI event is triggered the NMICNT will start decrementing. When NMICNT reaches zero, the device is Reset if the corresponding WDT or DMT flag is not cleared in SW. This NMI reset counter is only applicable to these two specific NMI events. The NMICNT cannot be updated or changed once in the NMI interrupt service routine by SW.

**Note:** The system unlock sequence must be performed before the SWRST bit is written. Refer to the **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the "PIC32 Family Reference Manual" for details.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 5-3: RNMICON: NON-MASKABLE INTERRUPT (NMI) CONTROL REGISTER

bit 17 **CF:** Clock Fail Detect bit

1 = FSCM has detected clock failure and caused an NMI

0 = FSCM has not detected clock failure

**Note:** On a clock fail event if enabled by the FCKSM<1:0> bits (DEVCFG1<15:14>) = '0b11, this bit and the OSCCON<CF> bit will be set. The user software must clear both the bits inside the CF NMI before attempting to exit the ISR. Software or hardware settings of the CF bit (OSCCON<3>) will cause a CF NMI event and an automatic clock switch to the Backup FRC (BFRC) provided the FCKSM<1:0> = '0b11. Unlike the CF bit (OSCCON<3>), software or hardware settings of the CF bit (RNMICON<17>) will cause a CF NMI event but will not cause a clock switch to the BFRC. After a Clock Fail event, a successful user software clock switch if implemented, hardware will automatically clear the CF bit (RNMICON<17>), but not the CF bit (OSCCON<3>). The CF bit (OSCCON<3>) must be cleared by software using the OSCCON register unlock procedure.

bit 16 **Unimplemented:** Read as '0'

bit 15-0 **NMICNT<15:0>:** NMI Reset Counter Value bits

These bits specify the reload value used by the NMI reset counter.

11111111-00000001 = Number of SYSCLK cycles before a device Reset occurs<sup>(1)</sup>

00000000 = No delay between NMI assertion and device Reset event

**Note 1:** When a Watchdog Timer NMI event (when not in Sleep mode) or a Deadman Timer NMI event is triggered the NMICNT will start decrementing. When NMICNT reaches zero, the device is Reset if the corresponding WDT or DMT flag is not cleared in SW. This NMI reset counter is only applicable to these two specific NMI events. The NMICNT cannot be updated or changed once in the NMI interrupt service routine by SW.

**Note:** The system unlock sequence must be performed before the SWRST bit is written. Refer to the **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the *"PIC32 Family Reference Manual"* for details.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 5-4: PWRCON: POWER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | RW-0          |
|           | —              | —              | —              | —              | —              | —              | —             | VREGS         |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-1 **Unimplemented:** Read as '0'

bit 0 **VREGS:** Internal Voltage Regulator Stand-by Enable bit

1 = Voltage regulator will remain active during Sleep

0 = Voltage regulator will go to Stand-by mode during Sleep

# PIC32MK GPG/MCJ with CAN FD Family

---

## 6.0 CPU EXCEPTIONS AND INTERRUPT CONTROLLER

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 8. “Interrupt Controller”** (DS60001108) and **Section 50. “CPU for Devices with MIPS32® microAptiv™ and M-Class Cores”** (DS60001192), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

PIC32MK GPG/MCJ with CAN FD Family of devices generate interrupt requests in response to interrupt events from peripheral modules. The Interrupt Controller module exists outside of the CPU and prioritizes the interrupt events before presenting them to the CPU.

The CPU handles interrupt events as part of the exception handling mechanism, which is described in **6.1 “CPU Exceptions”**.

The Interrupt Controller module includes the following features:

- Up to 216 interrupt sources and vectors with dedicated programmable offsets, eliminating the need for redirection
- Single and multi-vector mode operations
- Five external interrupts with edge polarity control
- Interrupt proximity timer
- Seven user-selectable priority levels for each vector
- Four user-selectable subpriority levels within each priority
- Seven shadow register sets that can be used for any priority level, eliminating software context switch and reducing interrupt latency
- Software can generate any interrupt

**Table 6-1** provides Interrupt Service routine (ISR) latency information.

**TABLE 6-1: ISR LATENCY INFORMATION**

| Condition  | Optimal Performance Configuration |                           |                         |                         |                                |                              | Recommended Settings for Optimal Interrupt Performance   | Comment   |
|--|-----------------------------------|---------------------------|-------------------------|-------------------------|--------------------------------|------------------------------|--|---|
|  | CP0 REGISTER 16, SELECT 0 <K0>    | PERCHEEN bit (CHECON<26>) | DCHEEN bit (CHECON<25>) | ICHEEN bit (CHECON<24>) | PREFEN<1:0> bits (CHECON<5:4>) | PFMWS <2:0> bits CHECON<2:0> | User source file ISR declaration/invocation.<br><b>Note:</b> The user is responsible for the ISR declaration for the fastest ISR latency response.   | Interrupt Latency (SYSCLK Cycles) (Time from interrupt event to first user source code instruction execution inside ISR). |
| Reset Values   | 3'b010                            | 1'b1                      | 1'b1                    | 1'b1                    | 2'b00                          | 3'b111                       | <pre>void __ISR(&lt;Vector Number n&gt;, ipl7auto) ISR(void) {     // "n" = Vector Number, see data sheet     // User ISR code }</pre>   | 257 Instr Cycles  |
| Recommended user optimized CPU and ISR Latency Settings (see <b>Note 2</b> ) | 3'b011                            | 1'b1                      | 1'b1                    | 1'b1                    | 2'b01                          | 3'b010                       | <b>(Refer to note 3)</b><br><pre>void __attribute__((interrupt(iplXsrs), at_vector(n), aligned(16))) isr () {     // "n"=Vector Number, see data sheet     // "X"=IPL 1-7     // User ISR code }</pre> | 30 + (7 – IPL) (Instr Cycles per interrupt <sup>(3)</sup> )   |

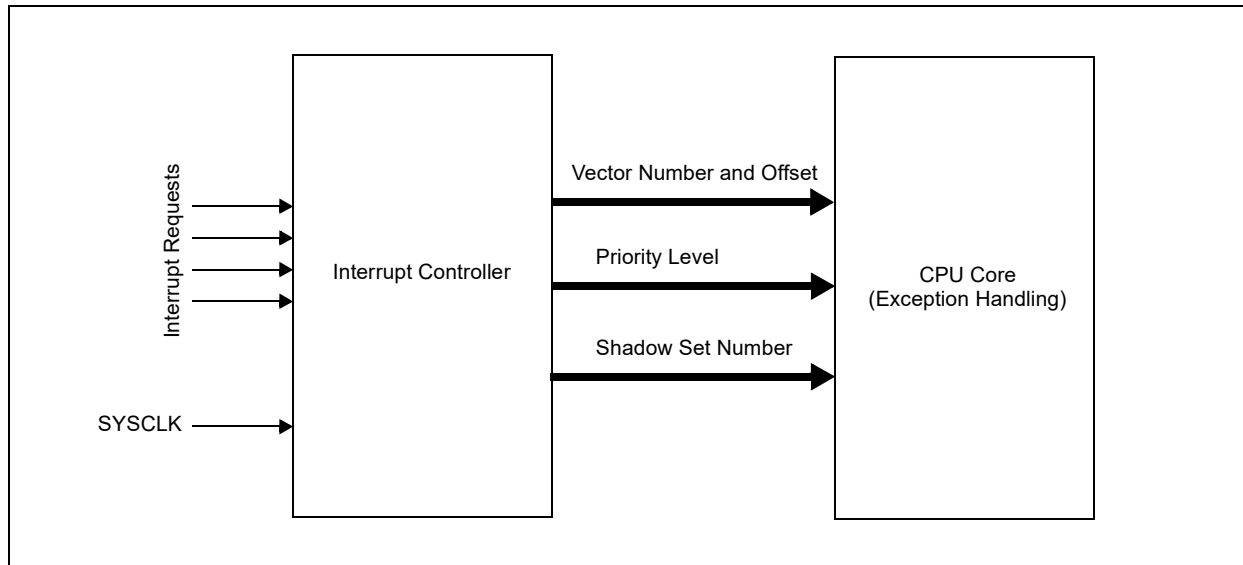
- Note 1:** The CPU ISR latency can cause unexpected behavior in high data rate peripherals when a high repetitive rate of CPU interrupts. For example, it is possible that if multiple interrupt sources occur simultaneously, or if a high-speed peripheral like ADC occurs faster than the CPU can read the results from the first original interrupt, then that data may be overwritten by the second interrupt. If the possibility exists in user application that the CPU servicing requirements are less than the combined sum of all possible overlapping interrupt rate specified above, then to avoid buffer overflows or data overwrites it is recommended to use the DMA to service the data and buffer instead of the CPU.
- 2:** For the best optimized CPU and ISR performance, to complete the optimization, the user application should define ISRs that use the “at vector” attribute as shown in [Table 6-1](#) In addition, if the ADC combined sum throughput rate of all the ADC modules in use is greater than (SYSCLK/43) = 2.8 Msps, it is recommended to use the ADC CPU early interrupt generation defined in the ADCxTIME and ADCEIENx registers. This will reduce the probability of the ADC results being overwritten by the next conversion before the CPU can read the previous ADC result if not using the DMA for ADC. Do not use the early interrupts if using the ADC in DMA mode.
- 3:** For optimal interrupt performance, (i.e. latency), the user should assign a unique shadow register set for each interrupt priority level by setting PRIS<PRIxSS> = IPL.

# PIC32MK GPG/MCJ with CAN FD Family

---

Figure 6-1 shows the block diagram for the Interrupt Controller and CPU exceptions.

**FIGURE 6-1: CPU EXCEPTIONS AND INTERRUPT CONTROLLER MODULE BLOCK DIAGRAM**



## 6.1 CPU Exceptions

CPU coprocessor 0 contains the logic for identifying and managing exceptions. Exceptions can be caused by a variety of sources, including boundary cases in data, external events or program errors. [Table 6-2](#) lists the exception types in order of priority.

**TABLE 6-2: MIPS32® microAptiv™ MCU CORE EXCEPTION TYPES**

| Exception Type<br>(In Order of<br>Priority) | Description  | Branches to                     | Status<br>Bits Set | Debug Bits<br>Set | EXCCODE         | XC32 Function Name                      |
|---|--|---------------------------------|--------------------|-------------------|-----------------|---|
| Highest Priority                            |  |                                 |                    |                   |                 |   |
| Reset                                       | Assertion $\overline{\text{MCLR}}$ or a Power-on Reset (POR).  | 0xBFC0_0000                     | BEV, ERL           | —                 | —               | <code>_on_reset</code>                  |
| Soft Reset                                  | Assertion of a software Reset.   | 0xBFC0_0000                     | BEV, SR,<br>ERL    | —                 | —               | <code>_on_reset</code>                  |
| DSS   | EJTAG debug single step.   | 0xBFC0_0480                     | —                  | DSS               | —               | —                                       |
| DINT  | EJTAG debug interrupt. Caused by the assertion of the external EJ_DINT input or by setting the EjtagBrk bit in the ECR register.   | 0xBFC0_0480                     | —                  | DINT              | —               | —                                       |
| NMI   | Assertion of NMI signal.   | 0xBFC0_0000                     | BEV, NMI,<br>ERL   | —                 | —               | <code>_nmi_handler</code>               |
| Interrupt                                   | Assertion of unmasked hardware or software interrupt signal.   | See <a href="#">Table 6-3</a> . | IPL<2:0>           | —                 | 0x00            | See <a href="#">Table 6-3</a> .         |
| Deferred Watch                              | Deferred watch (unmasked by $\text{K DM} \Rightarrow !(\text{K DM})$ transition).  | EBASE+0x180                     | WP, EXL            | —                 | 0x17            | <code>_general_exception_handler</code> |
| DIB   | EJTAG debug hardware instruction break matched.  | 0xBFC0_0480                     | —                  | DIB               | —               | —                                       |
| WATCH                                       | A reference to an address that is in one of the Watch registers (fetch).   | EBASE+0x180                     | EXL                | —                 | 0x17            | <code>_general_exception_handler</code> |
| AdEL  | Fetch address alignment error. Fetch reference to protected address.   | EBASE+0x180                     | EXL                | —                 | 0x04            | <code>_general_exception_handler</code> |
| IBE   | Instruction fetch bus error.   | EBASE+0x180                     | EXL                | —                 | 0x06            | <code>_general_exception_handler</code> |
| Instruction<br>Validity<br>Exceptions       | An instruction could not be completed because it was not allowed to access the required resources (Coprocessor Unusable) or was illegal (Reserved Instruction). If both exceptions occur on the same instruction, the Coprocessor Unusable Exception takes priority over the Reserved Instruction Exception. | EBASE+0x180                     | EXL                | —                 | 0x0A or<br>0x0B | <code>_general_exception_handler</code> |

**TABLE 6-2: MIPS32® microAptiv™ MCU CORE EXCEPTION TYPES (CONTINUED)**

| Exception Type<br>(In Order of<br>Priority) | Description  | Branches to | Status<br>Bits Set | Debug Bits<br>Set                           | EXCCODE   | XC32 Function Name         |
|---|--|-------------|--------------------|---|-----------|----------------------------|
| Execute<br>Exception                        | An instruction-based exception occurred: Integer overflow, trap, system call, breakpoint, floating point, or DSP ASE state disabled exception. | EBASE+0x180 | EXL                | —   | 0x08-0x0C | _general_exception_handler |
| Tr  | Execution of a trap (when trap condition is true).   | EBASE+0x180 | EXL                | —   | 0x0D      | _general_exception_handler |
| DDBL/DDBS                                   | EJTAG Data Address Break (address only) or EJTAG data value break on store (address + value).  | 0xBFC0_0480 | —                  | DDBL or<br>DDBS                             | —         | —                          |
| WATCH                                       | A reference to an address that is in one of the Watch registers (data).  | EBASE+0x180 | EXL                | —   | 0x17      | _general_exception_handler |
| AdEL  | Load address alignment error. User mode load reference to kernel address.  | EBASE+0x180 | EXL                | —   | 0x04      | _general_exception_handler |
| AdES  | Store address alignment error. User mode store to kernel address.  | EBASE+0x180 | EXL                | —   | 0x05      | _general_exception_handler |
| DBE   | Load or store bus error.   | EBASE+0x180 | EXL                | —   | 0x07      | _general_exception_handler |
| DDBL  | EJTAG data hardware breakpoint matched in load data compare.   | 0xBFC0_0480 | —                  | DDBL  | —         | —                          |
| CBrk  | EJTAG complex breakpoint.  | 0xBFC0_0480 | —                  | DIBIMPR,<br>DDBLIMPR,<br>and/or<br>DDBSIMPR | —         | —                          |
| Lowest Priority                             |  |             |                    |   |           |                            |



## 6.2 Interrupts

The PIC32MK GPG/MCJ with CAN FD Family of devices uses variable offsets for vector spacing. This allows the interrupt vector spacing to be configured according to application needs. A unique interrupt vector offset can be set for each vector using its associated OFFx register.

For additional information on the variable offset features, refer to **8.5.2 “Variable Offset”** in **Section 8. “Interrupt Controller”** (DS60001108) of the *“PIC32 Family Reference Manual”*.

[Table 6-3](#) provides the Interrupt IRQ, vector and bit location information.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name              | IRQ # | Vector #     | Interrupt Bit Location |          |             |              | Persistent Interrupt |
|---------------------------------|-------------------------------|-------|--------------|------------------------|----------|-------------|--------------|----------------------|
|                                 |                               |       |              | Flag                   | Enable   | Priority    | Sub-priority |                      |
| Highest Natural Order Priority  |                               |       |              |                        |          |             |              |                      |
| Core Timer Interrupt            | _CORE_TIMER_VECTOR            | 0     | OFF000<17:1> | IFS0<0>                | IEC0<0>  | IPC0<4:2>   | IPC0<1:0>    | No                   |
| Core Software Interrupt 0       | _CORE_SOFTWARE_0_VECTOR       | 1     | OFF001<17:1> | IFS0<1>                | IEC0<1>  | IPC0<12:10> | IPC0<9:8>    | No                   |
| Core Software Interrupt 1       | _CORE_SOFTWARE_1_VECTOR       | 2     | OFF002<17:1> | IFS0<2>                | IEC0<2>  | IPC0<20:18> | IPC0<17:16>  | No                   |
| External Interrupt 0            | _EXTERNAL_0_VECTOR            | 3     | OFF003<17:1> | IFS0<3>                | IEC0<3>  | IPC0<28:26> | IPC0<25:24>  | No                   |
| Timer1                          | _TIMER_1_VECTOR               | 4     | OFF004<17:1> | IFS0<4>                | IEC0<4>  | IPC1<4:2>   | IPC1<1:0>    | No                   |
| Input Capture 1 Error           | _INPUT_CAPTURE_1_ERROR_VECTOR | 5     | OFF005<17:1> | IFS0<5>                | IEC0<5>  | IPC1<12:10> | IPC1<9:8>    | Yes                  |
| Input Capture 1                 | _INPUT_CAPTURE_1_VECTOR       | 6     | OFF006<17:1> | IFS0<6>                | IEC0<6>  | IPC1<20:18> | IPC1<17:16>  | Yes                  |
| Output Compare 1                | _OUTPUT_COMPARE_1_VECTOR      | 7     | OFF007<17:1> | IFS0<7>                | IEC0<7>  | IPC1<28:26> | IPC1<25:24>  | No                   |
| External Interrupt 1            | _EXTERNAL_1_VECTOR            | 8     | OFF008<17:1> | IFS0<8>                | IEC0<8>  | IPC2<4:2>   | IPC2<1:0>    | No                   |
| Timer2                          | _TIMER_2_VECTOR               | 9     | OFF009<17:1> | IFS0<9>                | IEC0<9>  | IPC2<12:10> | IPC2<9:8>    | No                   |
| Input Capture 2 Error           | _INPUT_CAPTURE_2_ERROR_VECTOR | 10    | OFF010<17:1> | IFS0<10>               | IEC0<10> | IPC2<20:18> | IPC2<17:16>  | Yes                  |
| Input Capture 2                 | _INPUT_CAPTURE_2_VECTOR       | 11    | OFF011<17:1> | IFS0<11>               | IEC0<11> | IPC2<28:26> | IPC2<25:24>  | Yes                  |
| Output Compare 2                | _OUTPUT_COMPARE_2_VECTOR      | 12    | OFF012<17:1> | IFS0<12>               | IEC0<12> | IPC3<4:2>   | IPC3<1:0>    | No                   |
| External Interrupt 2            | _EXTERNAL_2_VECTOR            | 13    | OFF013<17:1> | IFS0<13>               | IEC0<13> | IPC3<12:10> | IPC3<9:8>    | No                   |
| Timer3                          | _TIMER_3_VECTOR               | 14    | OFF014<17:1> | IFS0<14>               | IEC0<14> | IPC3<20:18> | IPC3<17:16>  | No                   |
| Input Capture 3 Error           | _INPUT_CAPTURE_3_ERROR_VECTOR | 15    | OFF015<17:1> | IFS0<15>               | IEC0<15> | IPC3<28:26> | IPC3<25:24>  | Yes                  |
| Input Capture 3                 | _INPUT_CAPTURE_3_VECTOR       | 16    | OFF016<17:1> | IFS0<16>               | IEC0<16> | IPC4<4:2>   | IPC4<1:0>    | Yes                  |
| Output Compare 3                | _OUTPUT_COMPARE_3_VECTOR      | 17    | OFF017<17:1> | IFS0<17>               | IEC0<17> | IPC4<12:10> | IPC4<9:8>    | No                   |
| External Interrupt 3            | _EXTERNAL_3_VECTOR            | 18    | OFF018<17:1> | IFS0<18>               | IEC0<18> | IPC4<20:18> | IPC4<17:16>  | No                   |
| Timer4                          | _TIMER_4_VECTOR               | 19    | OFF019<17:1> | IFS0<19>               | IEC0<19> | IPC4<28:26> | IPC4<25:24>  | No                   |
| Input Capture 4 Error           | _INPUT_CAPTURE_4_ERROR_VECTOR | 20    | OFF020<17:1> | IFS0<20>               | IEC0<20> | IPC5<4:2>   | IPC5<1:0>    | Yes                  |
| Input Capture 4                 | _INPUT_CAPTURE_4_VECTOR       | 21    | OFF021<17:1> | IFS0<21>               | IEC0<21> | IPC5<12:10> | IPC5<9:8>    | Yes                  |
| Output Compare 4                | _OUTPUT_COMPARE_4_VECTOR      | 22    | OFF022<17:1> | IFS0<22>               | IEC0<22> | IPC5<20:18> | IPC5<17:16>  | No                   |
| External Interrupt 4            | _EXTERNAL_4_VECTOR            | 23    | OFF023<17:1> | IFS0<23>               | IEC0<23> | IPC5<28:26> | IPC5<25:24>  | No                   |
| Timer5                          | _TIMER_5_VECTOR               | 24    | OFF024<17:1> | IFS0<24>               | IEC0<24> | IPC6<4:2>   | IPC6<1:0>    | No                   |
| Input Capture 5 Error           | _INPUT_CAPTURE_5_ERROR_VECTOR | 25    | OFF025<17:1> | IFS0<25>               | IEC0<25> | IPC6<12:10> | IPC6<9:8>    | Yes                  |
| Input Capture 5                 | _INPUT_CAPTURE_5_VECTOR       | 26    | OFF026<17:1> | IFS0<26>               | IEC0<26> | IPC6<20:18> | IPC6<17:16>  | Yes                  |
| Output Compare 5                | _OUTPUT_COMPARE_5_VECTOR      | 27    | OFF027<17:1> | IFS0<27>               | IEC0<27> | IPC6<28:26> | IPC6<25:24>  | No                   |

**Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.

- 2:** This interrupt source is not available on 48-pin devices.  
**3:** This interrupt source is ONLY available on “MC” variants of the device.  
**4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name        | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|---------------------------------|-------------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                 |                         |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| Reserved                        | —                       | 28    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                       | 29    | —            | —                      | —        | —            | —            | —                    |
| Real Time Clock                 | _RTCC_VECTOR            | 30    | OFF030<17:1> | IFS0<30>               | IEC0<30> | IPC7<20:18>  | IPC7<17:16>  | Yes                  |
| Flash Control Event             | _FLASH_CONTROL_VECTOR   | 31    | OFF031<17:1> | IFS0<31>               | IEC0<31> | IPC7<28:26>  | IPC7<25:24>  | No                   |
| Comparator 1 Interrupt          | _COMPARATOR_1_VECTOR    | 32    | OFF032<17:1> | IFS1<0>                | IEC1<0>  | IPC8<4:2>    | IPC8<1:0>    | No                   |
| Comparator 2 Interrupt          | _COMPARATOR_2_VECTOR    | 33    | OFF033<17:1> | IFS1<1>                | IEC1<1>  | IPC8<12:10>  | IPC8<9:8>    | No                   |
| Reserved                        | —                       | 34    | —            | —                      | —        | —            | —            | —                    |
| SPI1 Fault                      | _SPI1_FAULT_VECTOR      | 35    | OFF035<17:1> | IFS1<3>                | IEC1<3>  | IPC8<28:26>  | IPC8<25:24>  | Yes                  |
| SPI1 Receive Done               | _SPI1_RX_VECTOR         | 36    | OFF036<17:1> | IFS1<4>                | IEC1<4>  | IPC9<4:2>    | IPC9<1:0>    | Yes                  |
| SPI1 Transfer Done              | _SPI1_TX_VECTOR         | 37    | OFF037<17:1> | IFS1<5>                | IEC1<5>  | IPC9<12:10>  | IPC9<9:8>    | Yes                  |
| UART1 Fault                     | _UART1_FAULT_VECTOR     | 38    | OFF038<17:1> | IFS1<6>                | IEC1<6>  | IPC9<20:18>  | IPC9<17:16>  | Yes                  |
| UART1 Receive Done              | _UART1_RX_VECTOR        | 39    | OFF039<17:1> | IFS1<7>                | IEC1<7>  | IPC9<28:26>  | IPC9<25:24>  | Yes                  |
| UART1 Transfer Done             | _UART1_TX_VECTOR        | 40    | OFF040<17:1> | IFS1<8>                | IEC1<8>  | IPC10<4:2>   | IPC10<1:0>   | Yes                  |
| I2C1 Bus Collision Event        | _I2C1_BUS_VECTOR        | 41    | OFF041<17:1> | IFS1<9>                | IEC1<9>  | IPC9<12:10>  | IPC9<9:8>    | Yes                  |
| I2C1 Slave Event                | _I2C1_SLAVE_VECTOR      | 42    | OFF042<17:1> | IFS1<10>               | IEC1<10> | IPC9<20:18>  | IPC9<17:16>  | Yes                  |
| I2C1 Master Event               | _I2C1_MASTER_VECTOR     | 43    | OFF043<17:1> | IFS1<11>               | IEC1<11> | IPC10<28:26> | IPC10<25:24> | No                   |
| PORTA Input Change Interrupt    | _CHANGE_NOTICE_A_VECTOR | 44    | OFF044<17:1> | IFS1<12>               | IEC1<12> | IPC11<4:2>   | IPC11<1:0>   | Yes                  |
| PORTB Input Change Interrupt    | _CHANGE_NOTICE_B_VECTOR | 45    | OFF045<17:1> | IFS1<13>               | IEC1<13> | IPC11<12:10> | IPC11<9:8>   | Yes                  |
| PORTC Input Change Interrupt    | _CHANGE_NOTICE_C_VECTOR | 46    | OFF046<17:1> | IFS1<14>               | IEC1<14> | IPC11<20:18> | IPC11<17:16> | Yes                  |
| PORTD Input Change Interrupt    | _CHANGE_NOTICE_D_VECTOR | 47    | OFF047<17:1> | IFS1<15>               | IEC1<15> | IPC11<28:26> | IPC11<25:24> | Yes                  |
| PORTE Input Change Interrupt    | _CHANGE_NOTICE_E_VECTOR | 48    | OFF048<17:1> | IFS1<16>               | IEC1<16> | IPC12<4:2>   | IPC12<1:0>   | Yes                  |
| PORTF Input Change Interrupt    | _CHANGE_NOTICE_F_VECTOR | 49    | OFF049<17:1> | IFS1<17>               | IEC1<17> | IPC12<12:10> | IPC12<9:8>   | Yes                  |
| PORTG Input Change Interrupt    | _CHANGE_NOTICE_G_VECTOR | 50    | OFF050<17:1> | IFS1<18>               | IEC1<18> | IPC12<20:18> | IPC12<17:16> | Yes                  |
| Reserved                        | —                       | 51    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                       | 52    | —            | —                      | —        | —            | —            | —                    |

- Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.
- 2:** This interrupt source is not available on 48-pin devices.
- 3:** This interrupt source is ONLY available on “MC” variants of the device.
- 4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name              | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|---------------------------------|-------------------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                 |                               |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| SPI2 Fault                      | _SPI2_FAULT_VECTOR            | 53    | OFF053<17:1> | IFS1<21>               | IEC1<21> | IPC13<12:10> | IPC13<9:8>   | Yes                  |
| SPI2 Receive Done               | _SPI2_RX_VECTOR               | 54    | OFF054<17:1> | IFS1<22>               | IEC1<22> | IPC13<20:18> | IPC13<17:16> | Yes                  |
| SPI2 Transfer Done              | _SPI2_TX_VECTOR               | 55    | OFF055<17:1> | IFS1<23>               | IEC1<23> | IPC13<28:26> | IPC13<25:24> | Yes                  |
| UART2 Fault                     | _UART2_FAULT_VECTOR           | 56    | OFF056<17:1> | IFS1<24>               | IEC1<24> | IPC14<4:2>   | IPC14<1:0>   | Yes                  |
| UART2 Receive Done              | _UART2_RX_VECTOR              | 57    | OFF057<17:1> | IFS1<25>               | IEC1<25> | IPC14<12:10> | IPC14<9:8>   | Yes                  |
| UART2 Transfer Done             | _UART2_TX_VECTOR              | 58    | OFF058<17:1> | IFS1<26>               | IEC1<26> | IPC14<20:18> | IPC14<17:16> | Yes                  |
| I2C2 Bus Collision Event        | _I2C2_BUS_VECTOR              | 59    | OFF059<17:1> | IFS1<27>               | IEC1<27> | IPC14<28:26> | IPC14<25:24> | Yes                  |
| I2C2 Slave Event                | _I2C2_SLAVE_VECTOR            | 60    | OFF060<17:1> | IFS1<28>               | IEC1<28> | IPC15<4:2>   | IPC15<1:0>   | Yes                  |
| I2C2 Master Event               | _I2C2_MASTER_VECTOR           | 61    | OFF061<17:1> | IFS1<29>               | IEC1<29> | IPC15<12:10> | IPC15<9:8>   | Yes                  |
| Reserved                        | —                             | 62    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 63    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 64    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 65    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 66    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 67    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 68    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 69    | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                             | 70    | —            | —                      | —        | —            | —            | —                    |
| CTMU Interrupt                  | _CTMU_VECTOR                  | 71    | OFF071<17:1> | IFS2<7>                | IEC2<7>  | IPC17<28:26> | IPC17<25:24> | Yes                  |
| DMA Channel 0                   | _DMA0_VECTOR                  | 72    | OFF072<17:1> | IFS2<8>                | IEC2<8>  | IPC18<4:2>   | IPC18<1:0>   | Yes                  |
| DMA Channel 1                   | _DMA1_VECTOR                  | 73    | OFF073<17:1> | IFS2<9>                | IEC2<9>  | IPC18<12:10> | IPC18<9:8>   | Yes                  |
| DMA Channel 2                   | _DMA2_VECTOR                  | 74    | OFF074<17:1> | IFS2<10>               | IEC2<10> | IPC18<20:18> | IPC18<17:16> | Yes                  |
| DMA Channel 3                   | _DMA3_VECTOR                  | 75    | OFF075<17:1> | IFS2<11>               | IEC2<11> | IPC18<28:26> | IPC18<25:24> | Yes                  |
| Timer6                          | _TIMER_6_VECTOR               | 76    | OFF076<17:1> | IFS2<12>               | IEC2<12> | IPC19<4:2>   | IPC19<1:0>   | Yes                  |
| Input Capture 6 Error           | _INPUT_CAPTURE_6_ERROR_VECTOR | 77    | OFF077<17:1> | IFS2<13>               | IEC2<13> | IPC19<12:10> | IPC19<9:8>   | Yes                  |
| Input Capture 6                 | _INPUT_CAPTURE_6_VECTOR       | 78    | OFF078<17:1> | IFS2<14>               | IEC2<14> | IPC19<20:18> | IPC19<17:16> | Yes                  |
| Output Compare 6                | _OUTPUT_COMPARE_6_VECTOR      | 79    | OFF079<17:1> | IFS2<15>               | IEC2<15> | IPC19<28:26> | IPC19<25:24> | Yes                  |
| Timer7                          | _TIMER_7_VECTOR               | 80    | OFF080<17:1> | IFS2<16>               | IEC2<16> | IPC20<4:2>   | IPC20<1:0>   | Yes                  |
| Input Capture 7 Error           | _INPUT_CAPTURE_7_ERROR_VECTOR | 81    | OFF081<17:1> | IFS2<17>               | IEC2<17> | IPC20<12:10> | IPC20<9:8>   | Yes                  |

**Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.

- 2:** This interrupt source is not available on 48-pin devices.  
**3:** This interrupt source is ONLY available on “MC” variants of the device.  
**4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name              | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|---------------------------------|-------------------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                 |                               |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| Input Capture 7                 | _INPUT_CAPTURE_7_VECTOR       | 82    | OFF082<17:1> | IFS2<18>               | IEC2<18> | IPC20<20:18> | IPC20<17:16> | Yes                  |
| Output Compare 7                | _OUTPUT_COMPARE_7_VECTOR      | 83    | OFF083<17:1> | IFS2<19>               | IEC2<19> | IPC20<28:26> | IPC20<25:24> | Yes                  |
| Timer8                          | _TIMER_8_VECTOR               | 84    | OFF084<17:1> | IFS2<20>               | IEC2<20> | IPC21<4:2>   | IPC21<1:0>   | Yes                  |
| Input Capture 8 Error           | _INPUT_CAPTURE_8_ERROR_VECTOR | 85    | OFF085<17:1> | IFS2<21>               | IEC2<21> | IPC21<12:10> | IPC21<9:8>   | Yes                  |
| Input Capture 8                 | _INPUT_CAPTURE_8_VECTOR       | 86    | OFF086<17:1> | IFS2<22>               | IEC2<22> | IPC21<20:18> | IPC21<17:16> | Yes                  |
| Output Compare 8                | _OUTPUT_COMPARE_8_VECTOR      | 87    | OFF087<17:1> | IFS2<23>               | IEC2<23> | IPC21<28:26> | IPC21<25:24> | Yes                  |
| Timer9                          | _TIMER_9_VECTOR               | 88    | OFF088<17:1> | IFS2<24>               | IEC2<24> | IPC22<4:2>   | IPC22<1:0>   | Yes                  |
| Input Capture 9 Error           | _INPUT_CAPTURE_9_ERROR_VECTOR | 89    | OFF089<17:1> | IFS2<25>               | IEC2<25> | IPC22<12:10> | IPC22<9:8>   | Yes                  |
| Input Capture 9                 | _INPUT_CAPTURE_9_VECTOR       | 90    | OFF090<17:1> | IFS2<26>               | IEC2<26> | IPC22<20:18> | IPC22<17:16> | Yes                  |
| Output Compare 9                | _OUTPUT_COMPARE_9_VECTOR      | 91    | OFF091<17:1> | IFS2<27>               | IEC2<27> | IPC22<28:26> | IPC22<25:24> | Yes                  |
| ADC Global Interrupt            | _ADC_VECTOR                   | 92    | OFF092<17:1> | IFS2<28>               | IEC2<28> | IPC23<4:2>   | IPC23<1:0>   | Yes                  |
| Reserved                        | —                             | 93    | —            | —                      | —        | —            | —            | —                    |
| ADC Digital Comparator 1        | _ADC_DC1_VECTOR               | 94    | OFF094<17:1> | IFS2<30>               | IEC2<30> | IPC23<20:18> | IPC23<17:16> | Yes                  |
| ADC Digital Comparator 2        | _ADC_DC2_VECTOR               | 95    | OFF095<17:1> | IFS2<31>               | IEC2<31> | IPC23<28:26> | IPC23<25:24> | Yes                  |
| ADC Digital Filter 1            | _ADC_DF1_VECTOR               | 96    | OFF096<17:1> | IFS3<0>                | IEC3<0>  | IPC24<4:2>   | IPC24<1:0>   | Yes                  |
| ADC Digital Filter 2            | _ADC_DF2_VECTOR               | 97    | OFF097<17:1> | IFS3<1>                | IEC3<1>  | IPC24<12:10> | IPC24<9:8>   | Yes                  |
| ADC Digital Filter 3            | _ADC_DF3_VECTOR               | 98    | OFF098<17:1> | IFS3<2>                | IEC3<2>  | IPC24<20:18> | IPC24<17:16> | Yes                  |
| ADC Digital Filter 4            | _ADC_DF4_VECTOR               | 99    | OFF099<17:1> | IFS3<3>                | IEC3<3>  | IPC24<28:26> | IPC24<25:24> | Yes                  |
| ADC Fault                       | _ADC_FAULT_VECTOR             | 100   | OFF100<17:1> | IFS3<4>                | IEC3<4>  | IPC25<4:2>   | IPC25<1:0>   | Yes                  |
| ADC End of Scan                 | _ADC_EOS_VECTOR               | 101   | OFF101<17:1> | IFS3<5>                | IEC3<5>  | IPC25<12:10> | IPC25<9:8>   | Yes                  |
| ADC Ready                       | _ADC_ARDY_VECTOR              | 102   | OFF102<17:1> | IFS3<6>                | IEC3<6>  | IPC25<20:18> | IPC25<17:16> | Yes                  |
| ADC Update Ready After Suspend  | _ADC_URDY_VECTOR              | 103   | OFF103<17:1> | IFS3<7>                | IEC3<7>  | IPC25<28:26> | IPC25<25:24> | Yes                  |
| ADC First Class Channels DMA    | _ADC_DMA_VECTOR               | 104   | OFF104<17:1> | IFS3<8>                | IEC3<8>  | IPC26<4:2>   | IPC26<1:0>   | No                   |
| ADC Early Group Interrupt       | _ADC_EARLY_VECTOR             | 105   | OFF105<17:1> | IFS3<9>                | IEC3<9>  | IPC26<12:10> | IPC26<9:8>   | Yes                  |
| ADC Data 0                      | _ADC_DATA0_VECTOR             | 106   | OFF106<17:1> | IFS3<10>               | IEC3<10> | IPC26<20:18> | IPC26<17:16> | Yes                  |
| ADC Data 1                      | _ADC_DATA1_VECTOR             | 107   | OFF107<17:1> | IFS3<11>               | IEC3<11> | IPC26<28:26> | IPC26<25:24> | Yes                  |
| ADC Data 2                      | _ADC_DATA2_VECTOR             | 108   | OFF108<17:1> | IFS3<12>               | IEC3<12> | IPC26<4:2>   | IPC27<1:0>   | Yes                  |
| ADC Data 3                      | _ADC_DATA3_VECTOR             | 109   | OFF109<17:1> | IFS3<13>               | IEC3<13> | IPC27<12:10> | IPC27<9:8>   | Yes                  |
| ADC Data 4                      | _ADC_DATA4_VECTOR             | 110   | OFF110<17:1> | IFS3<14>               | IEC3<14> | IPC27<20:18> | IPC27<17:16> | Yes                  |

- Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.
- 2:** This interrupt source is not available on 48-pin devices.
- 3:** This interrupt source is ONLY available on “MC” variants of the device.
- 4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name   | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|---------------------------------|--------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                 |                    |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| ADC Data 5                      | _ADC_DATA5_VECTOR  | 111   | OFF111<17:1> | IFS3<15>               | IEC3<15> | IPC27<28:26> | IPC27<25:24> | Yes                  |
| ADC Data 6                      | _ADC_DATA6_VECTOR  | 112   | OFF112<17:1> | IFS3<16>               | IEC3<16> | IPC28<4:2>   | IPC28<1:0>   | Yes                  |
| ADC Data 7                      | _ADC_DATA7_VECTOR  | 113   | OFF113<17:1> | IFS3<17>               | IEC3<17> | IPC28<12:10> | IPC28<9:8>   | Yes                  |
| ADC Data 8                      | _ADC_DATA8_VECTOR  | 114   | OFF114<17:1> | IFS3<18>               | IEC3<18> | IPC28<20:18> | IPC28<17:16> | Yes                  |
| ADC Data 9                      | _ADC_DATA9_VECTOR  | 115   | OFF115<17:1> | IFS3<19>               | IEC3<19> | IPC28<28:26> | IPC28<25:24> | Yes                  |
| ADC Data 10                     | _ADC_DATA10_VECTOR | 116   | OFF116<17:1> | IFS3<20>               | IEC3<20> | IPC29<4:2>   | IPC29<1:0>   | Yes                  |
| ADC Data 11                     | _ADC_DATA11_VECTOR | 117   | OFF117<17:1> | IFS3<21>               | IEC3<21> | IPC29<12:10> | IPC29<9:8>   | Yes                  |
| ADC Data 12 <sup>(2)</sup>      | _ADC_DATA12_VECTOR | 118   | OFF118<17:1> | IFS3<22>               | IEC3<22> | IPC29<20:18> | IPC29<17:16> | Yes                  |
| ADC Data 13 <sup>(2)</sup>      | _ADC_DATA13_VECTOR | 119   | OFF119<17:1> | IFS3<23>               | IEC3<23> | IPC29<28:26> | IPC29<25:24> | Yes                  |
| ADC Data 14 <sup>(2)</sup>      | _ADC_DATA14_VECTOR | 120   | OFF120<17:1> | IFS3<24>               | IEC3<24> | IPC30<4:2>   | IPC30<1:0>   | Yes                  |
| ADC Data 15 <sup>(2)</sup>      | _ADC_DATA15_VECTOR | 121   | OFF121<17:1> | IFS3<25>               | IEC3<25> | IPC30<12:10> | IPC30<9:8>   | Yes                  |
| ADC Data 16 <sup>(2)</sup>      | _ADC_DATA16_VECTOR | 122   | OFF122<17:1> | IFS3<26>               | IEC3<26> | IPC30<20:18> | IPC30<17:16> | Yes                  |
| ADC Data 17 <sup>(2)</sup>      | _ADC_DATA17_VECTOR | 123   | OFF123<17:1> | IFS3<27>               | IEC3<27> | IPC30<28:26> | IPC30<25:24> | Yes                  |
| ADC Data 18 <sup>(2)</sup>      | _ADC_DATA18_VECTOR | 124   | OFF124<17:1> | IFS3<28>               | IEC3<28> | IPC31<4:2>   | IPC31<1:0>   | Yes                  |
| ADC Data 19 <sup>(2)</sup>      | _ADC_DATA19_VECTOR | 125   | OFF125<17:1> | IFS3<29>               | IEC3<29> | IPC31<12:10> | IPC31<9:8>   | Yes                  |
| Reserved                        | —                  | 126   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 127   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 128   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 129   | —            | —                      | —        | —            | —            | —                    |
| ADC Data 24                     | _ADC_DATA24_VECTOR | 130   | OFF130<17:1> | IFS4<2>                | IEC4<2>  | IPC32<20:18> | IPC32<17:16> | Yes                  |
| ADC Data 25                     | _ADC_DATA25_VECTOR | 131   | OFF131<17:1> | IFS4<3>                | IEC4<3>  | IPC32<28:26> | IPC32<25:24> | Yes                  |
| ADC Data 26                     | _ADC_DATA26_VECTOR | 132   | OFF132<17:1> | IFS4<4>                | IEC4<4>  | IPC33<4:2>   | IPC33<1:0>   | Yes                  |
| ADC Data 27                     | _ADC_DATA27_VECTOR | 133   | OFF133<17:1> | IFS4<5>                | IEC4<5>  | IPC33<12:10> | IPC33<9:8>   | Yes                  |
| Reserved                        | —                  | 134   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 135   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 136   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 137   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 138   | —            | —                      | —        | —            | —            | —                    |
| Reserved                        | —                  | 139   | —            | —                      | —        | —            | —            | —                    |

- Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.
- 2:** This interrupt source is not available on 48-pin devices.
- 3:** This interrupt source is ONLY available on “MC” variants of the device.
- 4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup>      | XC32 Vector Name     | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|--------------------------------------|----------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                      |                      |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| Reserved                             | —                    | 140   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 141   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 142   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 143   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 144   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 145   | —            | —                      | —        | —            | —            | —                    |
| ADC Data 40 <sup>(2)</sup>           | _ADC_DATA40_VECTOR   | 146   | OFF146<17:1> | IFS4<18>               | IEC4<18> | IPC36<20:18> | IPC36<17:16> | Yes                  |
| ADC Data 41 <sup>(2)</sup>           | _ADC_DATA41_VECTOR   | 147   | OFF147<17:1> | IFS4<19>               | IEC4<19> | IPC36<28:26> | IPC36<25:24> | Yes                  |
| Reserved                             | —                    | 148   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 149   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 150   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 151   | —            | —                      | —        | —            | —            | —                    |
| ADC Data 46 <sup>(2)</sup>           | _ADC_DATA46_VECTOR   | 152   | OFF152<17:1> | IFS4<24>               | IEC4<24> | IPC38<4:2>   | IPC38<1:0>   | Yes                  |
| ADC Data 47 <sup>(2)</sup>           | _ADC_DATA47_VECTOR   | 153   | OFF153<17:1> | IFS4<25>               | IEC4<25> | IPC38<12:10> | IPC38<9:8>   | Yes                  |
| ADC Data 48                          | _ADC_DATA48_VECTOR   | 154   | OFF154<17:1> | IFS4<26>               | IEC4<26> | IPC38<20:18> | IPC38<17:16> | Yes                  |
| ADC Data 49                          | _ADC_DATA49_VECTOR   | 155   | OFF155<17:1> | IFS4<27>               | IEC4<27> | IPC38<28:26> | IPC38<25:24> | Yes                  |
| ADC Data 50                          | _ADC_DATA50_VECTOR   | 156   | OFF156<17:1> | IFS4<28>               | IEC4<28> | IPC39<4:2>   | IPC39<1:0>   | Yes                  |
| Reserved                             | —                    | 157   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 158   | —            | —                      | —        | —            | —            | —                    |
| ADC Data 53                          | _ADC_DATA53_VECTOR   | 159   | OFF159<17:1> | IFS4<31>               | IEC4<31> | IPC39<28:26> | IPC39<25:24> | Yes                  |
| Comparator 3 Interrupt               | _COMPARATOR_3_VECTOR | 160   | OFF160<17:1> | IFS5<0>                | IEC5<0>  | IPC40<4:2>   | IPC40<1:0>   | No                   |
| Comparator 4 Interrupt               | _COMPARATOR_4_VECTOR | 161   | OFF161<17:1> | IFS5<1>                | IEC5<1>  | IPC40<12:10> | IPC40<9:8>   | No                   |
| Comparator 5 Interrupt               | _COMPARATOR_5_VECTOR | 162   | OFF162<17:1> | IFS5<2>                | IEC5<2>  | IPC40<20:18> | IPC40<17:16> | No                   |
| Reserved                             | —                    | 163   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 164   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 165   | —            | —                      | —        | —            | —            | —                    |
| Reserved                             | —                    | 166   | —            | —                      | —        | —            | —            | —                    |
| CAN1 Global Interrupt <sup>(4)</sup> | _CAN1_VECTOR         | 167   | OFF167<17:1> | IFS5<7>                | IEC5<7>  | IPC41<28:26> | IPC41<25:24> | Yes                  |
| Reserved                             | —                    | 168   | —            | —                      | —        | —            | —            | —                    |

**Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.

- 2:** This interrupt source is not available on 48-pin devices.  
**3:** This interrupt source is ONLY available on “MC” variants of the device.  
**4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup>  | XC32 Vector Name | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|--|------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|  |                  |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| QE11 Interrupt <sup>(3)</sup>  | _QE11_VECTOR     | 169   | OFF169<17:1> | IFS5<9>                | IEC5<9>  | IPC42<12:10> | IPC42<9:8>   | Yes                  |
| QE12 Interrupt <sup>(3)</sup>  | _QE12_VECTOR     | 170   | OFF170<17:1> | IFS5<10>               | IEC5<10> | IPC42<20:18> | IPC42<17:16> | Yes                  |
| PWM Primary Event <sup>(3)</sup>   | _PWM_PRI_VECTOR  | 171   | OFF171<17:1> | IFS5<11>               | IEC5<11> | IPC42<28:26> | IPC42<25:24> | No                   |
| PWM Sec Event <sup>(3)</sup>   | _PWM_SEC_VECTOR  | 172   | OFF172<17:1> | IFS5<12>               | IEC5<12> | IPC43<4:2>   | IPC43<1:0>   | No                   |
| PWM1 Combined Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM1_VECTOR     | 173   | OFF173<17:1> | IFS5<13>               | IEC5<13> | IPC43<12:10> | IPC43<9:8>   | No                   |
| PWM2 Combined Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM2_VECTOR     | 174   | OFF174<17:1> | IFS5<14>               | IEC5<14> | IPC43<20:18> | IPC43<17:16> | No                   |
| PWM3 Combined Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM3_VECTOR     | 175   | OFF175<17:1> | IFS5<15>               | IEC5<15> | IPC43<28:26> | IPC43<25:24> | No                   |
| PWM4 Combined Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM4_VECTOR     | 176   | OFF176<17:1> | IFS5<16>               | IEC5<16> | IPC44<4:2>   | IPC44<1:0>   | No                   |
| PWM5 Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup>          | _PWM5_VECTOR     | 177   | OFF177<17:1> | IFS5<17>               | IEC5<17> | IPC44<12:10> | IPC44<9:8>   | No                   |
| PWM6 Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup>          | _PWM6_VECTOR     | 178   | OFF178<17:1> | IFS5<18>               | IEC5<18> | IPC44<20:18> | IPC44<17:16> | No                   |
| Reserved   | —                | 179   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 180   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 181   | —            | —                      | —        | —            | —            | —                    |
| DMA Channel 4  | _DMA4_VECTOR     | 182   | OFF182<17:1> | IFS5<22>               | IEC5<22> | IPC45<20:18> | IPC45<17:16> | Yes                  |
| DMA Channel 5  | _DMA5_VECTOR     | 183   | OFF183<17:1> | IFS5<23>               | IEC5<23> | IPC45<28:26> | IPC45<25:24> | Yes                  |
| DMA Channel 6  | _DMA6_VECTOR     | 184   | OFF184<17:1> | IFS5<24>               | IEC5<24> | IPC46<4:2>   | IPC46<1:0>   | Yes                  |
| DMA Channel 7  | _DMA7_VECTOR     | 185   | OFF185<17:1> | IFS5<25>               | IEC5<25> | IPC46<12:10> | IPC46<9:8>   | Yes                  |
| Reserved   | —                | 186   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 187   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 188   | —            | —                      | —        | —            | —            | —                    |
| QE13 Interrupt <sup>(3)</sup>  | _QE13_VECTOR     | 189   | OFF189<17:1> | IFS5<29>               | IEC5<29> | IPC47<12:10> | IPC47<9:8>   | Yes                  |
| Reserved   | —                | 190   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 191   | —            | —                      | —        | —            | —            | —                    |
| Reserved   | —                | 192   | —            | —                      | —        | —            | —            | —                    |

**Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.

- 2:** This interrupt source is not available on 48-pin devices.  
**3:** This interrupt source is ONLY available on “MC” variants of the device.  
**4:** Only available on “MC” variants.



**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup> | XC32 Vector Name | IRQ # | Vector # | Interrupt Bit Location |        |          |              | Persistent Interrupt |
|---------------------------------|------------------|-------|----------|------------------------|--------|----------|--------------|----------------------|
|                                 |                  |       |          | Flag                   | Enable | Priority | Sub-priority |                      |
| Reserved                        | —                | 193   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 194   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 195   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 196   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 197   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 198   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 199   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 200   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 201   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 202   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 203   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 204   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 205   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 206   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 207   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 208   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 209   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 210   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 211   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 212   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 213   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 214   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 215   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 216   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 217   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 218   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 219   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 220   | —        | —                      | —      | —        | —            | —                    |
| Reserved                        | —                | 221   | —        | —                      | —      | —        | —            | —                    |

- Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.
- 2:** This interrupt source is not available on 48-pin devices.
- 3:** This interrupt source is ONLY available on “MC” variants of the device.
- 4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup>                                       | XC32 Vector Name              | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|---|-------------------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|   |                               |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| Reserved  | —                             | 222   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 223   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 224   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 225   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 226   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 227   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 228   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 229   | —            | —                      | —        | —            | —            | —                    |
| System Bus Protection Violation                                       | _SYSTEM_BUS_PROTECTION_VECTOR | 230   | OFF230<17:1> | IFS7<6>                | IEC7<6>  | IPC57<20:18> | IPC57<17:16> | Yes                  |
| Reserved  | —                             | 231   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 232   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 233   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 234   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 235   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 236   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 237   | —            | —                      | —        | —            | —            | —                    |
| PWM7 Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM7_VECTOR                  | 238   | OFF238<17:1> | IFS7<14>               | IEC7<14> | IPC59<20:18> | IPC59<17:16> | No                   |
| PWM8 Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM8_VECTOR                  | 239   | OFF239<17:1> | IFS7<15>               | IEC7<15> | IPC59<28:26> | IPC59<25:24> | No                   |
| PWM9 Interrupt (Period, Fault, Trigger, Current-Limit) <sup>(3)</sup> | _PWM9_VECTOR                  | 240   | OFF240<17:1> | IFS7<16>               | IEC7<16> | IPC60<4:2>   | IPC60<1:0>   | No                   |
| Reserved  | —                             | 241   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 242   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 243   | —            | —                      | —        | —            | —            | —                    |
| Reserved  | —                             | 244   | —            | —                      | —        | —            | —            | —                    |
| ADC Digital Comparator 3  | _ADC_DC3_VECTOR               | 245   | OFF245<17:1> | IFS7<21>               | IEC7<21> | IPC61<12:10> | IPC61<9:8>   | Yes                  |
| ADC Digital Comparator 4  | _ADC_DC4_VECTOR               | 246   | OFF246<17:1> | IFS7<22>               | IEC7<22> | IPC61<20:18> | IPC61<17:16> | Yes                  |
| Prefetch Cache Event  | _PCACHE_VECTOR                | 247   | OFF247<17:1> | IFS7<23>               | IEC7<23> | IPC61<28:26> | IPC61<25:24> | Yes                  |

**Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.

- 2:** This interrupt source is not available on 48-pin devices.  
**3:** This interrupt source is ONLY available on “MC” variants of the device.  
**4:** Only available on “MC” variants.

**TABLE 6-3: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

| Interrupt Source <sup>(1)</sup>    | XC32 Vector Name             | IRQ # | Vector #     | Interrupt Bit Location |          |              |              | Persistent Interrupt |
|------------------------------------|------------------------------|-------|--------------|------------------------|----------|--------------|--------------|----------------------|
|                                    |                              |       |              | Flag                   | Enable   | Priority     | Sub-priority |                      |
| Reserved                           | —                            | 248   | —            | —                      | —        | —            | —            | —                    |
| CLC1                               | _CLC1_VECTOR                 | 249   | OFF249<17:1> | IFS7<25>               | IEC7<25> | IPC62<12:10> | IPC62<9:8>   | Yes                  |
| CLC2                               | _CLC2_VECTOR                 | 250   | OFF250<17:1> | IFS7<26>               | IEC7<26> | IPC62<20:18> | IPC62<17:16> | Yes                  |
| CLC3                               | _CLC3_VECTOR                 | 251   | OFF251<17:1> | IFS7<27>               | IEC7<27> | IPC62<28:26> | IPC62<25:24> | Yes                  |
| CLC4                               | _CLC4_VECTOR                 | 252   | OFF252<17:1> | IFS7<28>               | IEC7<28> | IPC63<4:2>   | IPC62<1:0>   | Yes                  |
| Reserved                           | —                            | 253   | —            | —                      | —        | —            | —            | —                    |
| Core Performance Counter Interrupt | _CORE_PERF_COUNT_VECTOR      | 254   | OFF254<17:1> | IFS7<30>               | IEC7<30> | IPC63<20:18> | IPC63<17:16> | —                    |
| Fast Debug Channel Interrupt       | _CORE_FAST_DEBUG_CHAN_VECTOR | 255   | OFF255<17:1> | IFS7<31>               | IEC7<31> | IPC63<28:26> | IPC63<25:24> | —                    |
| Lowest Natural Order Priority      |                              |       |              |                        |          |              |              |                      |

- Note 1:** Not all interrupt sources are available on all devices. See [TABLE 1: “PIC32MK Motor Control and General Purpose \(MC and GP\) Family Features”](#) for the list of available peripherals.
- 2:** This interrupt source is not available on 48-pin devices.
- 3:** This interrupt source is ONLY available on “MC” variants of the device.
- 4:** Only available on “MC” variants.

### 6.3 Interrupt Control Registers

**TABLE 6-4: INTERRUPT REGISTER MAP**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits          |          |          |            |             |            |          |           |                       |          |          |          |             |          |          |          | All Resets |
|-----------------------------|---------------------------------|-----------|---------------|----------|----------|------------|-------------|------------|----------|-----------|-----------------------|----------|----------|----------|-------------|----------|----------|----------|------------|
|                             |                                 |           | 31/15         | 30/14    | 29/13    | 28/12      | 27/11       | 26/10      | 25/9     | 24/8      | 23/7                  | 22/6     | 21/5     | 20/4     | 19/3        | 18/2     | 17/1     | 16/0     |            |
| 0000                        | INTCON                          | 31:16     | SWNMIKEY<7:0> |          |          |            |             |            |          |           | —                     | —        | —        | —        | —           | —        | —        | —        | 0000       |
|                             |                                 | 15:0      | —             | —        | —        | MVEC       | —           | TPC<2:0>   |          |           | —                     | —        | —        | INT4EP   | INT3EP      | INT2EP   | INT1EP   | INT0EP   | 0000       |
| 0010                        | PRISS                           | 31:16     | PRI7SS<3:0>   |          |          |            | PRI6SS<3:0> |            |          |           | PRI5SS<3:0>           |          |          |          | PRI4SS<3:0> |          |          |          | 0000       |
|                             |                                 | 15:0      | PRI3SS<3:0>   |          |          |            | PRI2SS<3:0> |            |          |           | PRI1SS<3:0>           |          |          |          | —           | —        | —        | —        | SS0        |
| 0020                        | INTSTAT                         | 31:16     | —             | —        | —        | —          | —           | —          | —        | —         | —                     | —        | —        | —        | —           | —        | —        | —        | 0000       |
|                             |                                 | 15:0      | —             | —        | —        | —          | —           | SRIPL<2:0> |          |           | SIRQ<7:0>             |          |          |          |             |          |          | 0000     |            |
| 0030                        | IPTMR                           | 31:16     | IPTMR<31:0>   |          |          |            |             |            |          |           |                       |          |          |          |             |          |          |          | 0000       |
|                             |                                 | 15:0      |               |          |          |            |             |            |          |           |                       |          |          |          |             |          |          |          | 0000       |
| 0040                        | IFS0 <sup>(4)</sup>             | 31:16     | FCEIF         | RTCCIF   | —        | —          | OC5IF       | IC5IF      | IC5EIF   | T5IF      | INT4IF                | OC4IF    | IC4IF    | IC4EIF   | T4IF        | INT3IF   | OC3IF    | IC3IF    | 0000       |
|                             |                                 | 15:0      | IC3EIF        | T3IF     | INT2IF   | OC2IF      | IC2IF       | IC2EIF     | T2IF     | INT1IF    | OC1IF                 | IC1IF    | IC1EIF   | T1IF     | INT0IF      | CS1IF    | CS0IF    | CTIF     | 0000       |
| 0050                        | IFS1 <sup>(4)</sup>             | 31:16     | —             | —        | —        | —          | —           | U2TXIF     | U2RXIF   | U2EIF     | SPI2TXIF              | SPI2RXIF | SPI2EIF  | —        | —           | CNGIF    | CNFIF    | CNEIF    | 0000       |
|                             |                                 | 15:0      | CNDIF         | CNCIF    | CNBIF    | CNAIF      | —           | —          | —        | U1TXIF    | U1RXIF                | U1EIF    | SPI1TXIF | SPI1RXIF | SPI1EIF     | —        | CMP2IF   | CMP1IF   | 0000       |
| 0060                        | IFS2 <sup>(4)</sup>             | 31:16     | AD1DC2IF      | AD1DC1IF | —        | AD11F      | OC9IF       | IC9IF      | IC9EIF   | T9IF      | OC8IF                 | IC8IF    | IC8EIF   | T8IF     | OC7IF       | IC7IF    | IC7EIF   | T7IF     | 0000       |
|                             |                                 | 15:0      | OC6IF         | IC6IF    | IC6EIF   | T6IF       | DMA3IF      | DMA2IF     | DMA1IF   | DMA0IF    | CTMUIF                | —        | —        | —        | —           | —        | —        | —        | 0000       |
| 0070                        | IFS3 <sup>(4)</sup>             | 31:16     | —             | —        | AD1D19IF | AD1D18IF   | AD1D17IF    | AD1D16IF   | AD1D15IF | AD1D14IF  | AD1D13IF              | AD1D12IF | AD1D11IF | AD1D10IF | AD1D9IF     | AD1D8IF  | AD1D7IF  | AD1D6IF  | 0000       |
|                             |                                 | 15:0      | AD1D5IF       | AD1D4IF  | AD1D3IF  | AD1D2IF    | AD1D1IF     | AD1D0IF    | AD1G1IF  | AD1FCBTIF | AD1RSIF               | AD1ARIF  | AD1EOSIF | AD1F1IF  | AD1DF4IF    | AD1DF3IF | AD1DF2IF | AD1DF1IF | 0000       |
| 0080                        | IFS4 <sup>(4)</sup>             | 31:16     | AD1D53IF      | AD1D52IF | —        | AD1D50IF   | AD1D49IF    | AD1D48IF   | AD1D47IF | AD1D46IF  | —                     | —        | —        | —        | AD1D41IF    | AD1D40IF | —        | —        | 0000       |
|                             |                                 | 15:0      | —             | —        | —        | —          | —           | —          | —        | —         | —                     | —        | AD1D27IF | AD1D26IF | AD1D25IF    | AD1D24IF | —        | —        | 0000       |
| 0090                        | IFS5 <sup>(4)</sup>             | 31:16     | —             | —        | QE13IF   | —          | —           | —          | DMA7IF   | DMA6IF    | DMA5IF                | DMA4IF   | —        | —        | —           | PWM6IF   | PWM5IF   | PWM4IF   | 0000       |
|                             |                                 | 15:0      | PWM3IF        | PWM2IF   | PWM1IF   | PWM SEVTIF | PWM PEVTIF  | QE12IF     | QE11IF   | —         | CAN1IF <sup>(3)</sup> | —        | —        | —        | —           | CMP5IF   | CMP4IF   | CMP3IF   | 0000       |
| 00A0                        | IFS6 <sup>(4)</sup>             | 31:16     | —             | —        | —        | —          | —           | —          | —        | —         | —                     | —        | —        | —        | —           | —        | —        | 0000     |            |
|                             |                                 | 15:0      | —             | —        | —        | —          | —           | —          | —        | —         | —                     | —        | —        | —        | —           | —        | —        | —        | 0000       |
| 00B0                        | IFS7 <sup>(4)</sup>             | 31:16     | —             | CPCIF    | —        | CLC4       | CLC3        | CLC2       | CLC1     | —         | —                     | AD1DC4IF | AD1DC3IF | —        | —           | —        | —        | PWM9IF   | 0000       |
|                             |                                 | 15:0      | PWM8IF        | PWM7IF   | —        | —          | —           | —          | —        | —         | —                     | —        | SBIF     | —        | —           | —        | —        | —        | 0000       |
| 00C0                        | IEC0                            | 31:16     | FCEIE         | RTCCIE   | —        | —          | OC5IE       | IC5IE      | IC5EIE   | T5IE      | INT4IE                | OC4IE    | IC4IE    | IC4EIE   | T4IE        | INT3IE   | OC3IE    | IC3IE    | 0000       |
|                             |                                 | 15:0      | IC3EIE        | T3IE     | INT2IE   | OC2IE      | IC2IE       | IC2EIE     | T2IE     | INT1IE    | OC1IE                 | IC1IE    | IC1EIE   | T1IE     | INT0IE      | CS1IE    | CS0IE    | CTIE     | 0000       |
| 00D0                        | IEC1                            | 31:16     | —             | —        | —        | —          | —           | U2TXIE     | U2RXIE   | U2EIE     | SPI2TXIE              | SPI2RXIE | SPI2EIE  | —        | —           | CNGIE    | CNFIE    | CNEIE    | 0000       |
|                             |                                 | 15:0      | CNDIE         | CNCIE    | CNBIE    | CNAIE      | —           | —          | —        | U1TXIE    | U1RXIE                | U1EIE    | SPI1TXIE | SPI1RXIE | SPI1EIE     | —        | CMP2IE   | CMP1IE   | 0000       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits     |          |          |            |              |          |          |              |                       |          |          |          |             |          |          |             | All Resets |      |
|-----------------------------|---------------------------------|-----------|----------|----------|----------|------------|--------------|----------|----------|--------------|-----------------------|----------|----------|----------|-------------|----------|----------|-------------|------------|------|
|                             |                                 |           | 31/15    | 30/14    | 29/13    | 28/12      | 27/11        | 26/10    | 25/9     | 24/8         | 23/7                  | 22/6     | 21/5     | 20/4     | 19/3        | 18/2     | 17/1     | 16/0        |            |      |
| 00E0                        | IEC2                            | 31:16     | AD1DC2IE | AD1DC1IE | -        | AD11E      | OC9IE        | IC9IE    | IC9EIE   | T9IE         | OC8IE                 | IC8IE    | IC8EIE   | T8IE     | OC7IE       | IC7IE    | IC7EIE   | T7IE        | 0000       |      |
|                             |                                 | 15:0      | OC6IE    | IC6IE    | IC6EIE   | T6IE       | DMA3IE       | DMA2IE   | DMA1IE   | DMA0IE       | CTMUIE                | —        | —        | —        | —           | —        | —        | —           | —          | 0000 |
| 00F0                        | IEC3                            | 31:16     | —        | —        | AD1D19IE | AD1D18IE   | AD1D17IE     | AD1D16IE | AD1D15IE | AD1D14IE     | AD1D13IE              | AD1D12IE | AD1D11IE | AD1D10IE | AD1D09IE    | AD1D08IE | AD1D07IE | AD1D06IE    | 0000       |      |
|                             |                                 | 15:0      | AD1D05IE | AD1D04IE | AD1D03IE | AD1D02IE   | AD1D01IE     | AD1D00IE | AD1G1IE  | AD1FCBTIE    | AD1RSIE               | AD1ARIE  | AD1EOSIE | AD1F1IE  | AD1DF4IE    | AD1DF3IE | AD1DF2IE | AD1DF1IE    | 0000       |      |
| 0100                        | IEC4                            | 31:16     | AD1D53IE | AD1D52IE | —        | AD1D50IE   | AD1D49IE     | AD1D48IE | AD1D47IE | AD1D46IE     | —                     | —        | —        | —        | AD1D27IE    | AD1D26IE | AD1D25IE | AD1D24IE    | —          | —    |
|                             |                                 | 15:0      | —        | —        | —        | —          | —            | —        | —        | —            | —                     | —        | —        | —        | —           | —        | —        | —           | —          | —    |
| 0110                        | IEC5                            | 31:16     | —        | —        | —        | —          | —            | —        | —        | DMA7IE       | DMA6IE                | DMA5IE   | DMA4IE   | —        | —           | —        | PWM6IE   | PWM5IE      | PWM4IE     | 0000 |
|                             |                                 | 15:0      | PWM3IE   | PWM2IE   | PWM1IE   | PWM SEVTIE | PWM PEVTIE   | QEI2IE   | QEI1IE   | —            | CAN1IE <sup>(3)</sup> | —        | —        | —        | —           | —        | CMP5IE   | CMP4IE      | CMP3IE     | 0000 |
| 0120                        | IEC6                            | 31:16     | —        | —        | —        | —          | —            | —        | —        | —            | —                     | —        | —        | —        | —           | —        | —        | —           | —          | 0000 |
|                             |                                 | 15:0      | —        | —        | —        | —          | —            | —        | —        | —            | —                     | —        | —        | —        | —           | —        | —        | —           | —          | —    |
| 0130                        | IEC7                            | 31:16     | —        | CPCIE    | —        | CLC4IE     | CLC3IE       | CLC2IE   | CLC1IE   | —            | —                     | AD1DC4IE | AD1DC3IE | —        | PWM12IE     | PWM11IE  | PWM10IE  | PWM9IE      | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | —            | —        | —        | —            | —                     | —        | —        | —        | —           | —        | —        | —           | —          | —    |
| 0140                        | IPC0                            | 31:16     | —        | —        | —        | —          | INT0IP<2:0>  | —        | —        | INT0IS<1:0>  | —                     | —        | —        | —        | CS1IP<2:0>  | —        | —        | CS1IS<1:0>  | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | CS0IP<2:0>   | —        | —        | CS0IS<1:0>   | —                     | —        | —        | —        | CTIP<2:0>   | —        | —        | CTIS<1:0>   | 0000       |      |
| 0150                        | IPC1                            | 31:16     | —        | —        | —        | —          | OC1IP<2:0>   | —        | —        | OC1IS<1:0>   | —                     | —        | —        | —        | IC1IP<2:0>  | —        | —        | IC1IS<1:0>  | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | IC1EIP<2:0>  | —        | —        | IC1EIS<1:0>  | —                     | —        | —        | —        | T1IP<2:0>   | —        | —        | T1IS<1:0>   | 0000       |      |
| 0160                        | IPC2                            | 31:16     | —        | —        | —        | —          | IC2IP<2:0>   | —        | —        | IC2IS<1:0>   | —                     | —        | —        | —        | IC2EIP<2:0> | —        | —        | IC2EIS<1:0> | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | T2IP<2:0>    | —        | —        | T2IS<1:0>    | —                     | —        | —        | —        | INT1IP<2:0> | —        | —        | INT1IS<1:0> | 0000       |      |
| 0170                        | IPC3                            | 31:16     | —        | —        | —        | —          | IC3EIP<2:0>  | —        | —        | IC3EIS<1:0>  | —                     | —        | —        | —        | T3IP<2:0>   | —        | —        | T3IS<1:0>   | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | INT2IP<2:0>  | —        | —        | INT2IS<1:0>  | —                     | —        | —        | —        | OC2IP<2:0>  | —        | —        | OC2IS<1:0>  | 0000       |      |
| 0180                        | IPC4                            | 31:16     | —        | —        | —        | —          | T4IP<2:0>    | —        | —        | T4IS<1:0>    | —                     | —        | —        | —        | INT3IP<2:0> | —        | —        | INT3IS<1:0> | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | OC3IP<2:0>   | —        | —        | OC3IS<1:0>   | —                     | —        | —        | —        | IC3IP<2:0>  | —        | —        | IC3IS<1:0>  | 0000       |      |
| 0190                        | IPC5                            | 31:16     | —        | —        | —        | —          | INT4IP<2:0>  | —        | —        | INT4IS<1:0>  | —                     | —        | —        | —        | OC4IP<2:0>  | —        | —        | OC4IS<1:0>  | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | IC4IP<2:0>   | —        | —        | IC4IS<1:0>   | —                     | —        | —        | —        | IC4EIP<2:0> | —        | —        | IC4EIS<1:0> | 0000       |      |
| 01A0                        | IPC6                            | 31:16     | —        | —        | —        | —          | OC5IP<2:0>   | —        | —        | OC5IS<1:0>   | —                     | —        | —        | —        | IC5IP<2:0>  | —        | —        | IC5IS<1:0>  | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | IC5EIP<2:0>  | —        | —        | IC5EIS<1:0>  | —                     | —        | —        | —        | T5IP<2:0>   | —        | —        | T5IS<1:0>   | 0000       |      |
| 01B0                        | IPC7                            | 31:16     | —        | —        | —        | —          | FCEIP<2:0>   | —        | —        | FCEIS<1:0>   | —                     | —        | —        | —        | RTCCIP<2:0> | —        | —        | RTCCIS<1:0> | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | —            | —        | —        | —            | —                     | —        | —        | —        | —           | —        | —        | —           | —          | 0000 |
| 01C0                        | IPC8                            | 31:16     | —        | —        | —        | —          | SPI1EIP<2:0> | —        | —        | SPI1EIS<1:0> | —                     | —        | —        | —        | —           | —        | —        | —           | 0000       |      |
|                             |                                 | 15:0      | —        | —        | —        | —          | CMP2IP<2:0>  | —        | —        | CMP2IS<1:0>  | —                     | —        | —        | —        | CMP1IP<2:0> | —        | —        | CMP1IS<1:0> | 0000       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - This bit is not available on 64-pin devices.
  - This bit is not available on devices without a CAN module.
  - The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |               |       |       |               |      |      |      |      |      |               |      | All Resets |               |      |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|---------------|-------|-------|---------------|------|------|------|------|------|---------------|------|------------|---------------|------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12         | 27/11 | 26/10 | 25/9          | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3          | 18/2 |            | 17/1          | 16/0 |      |
| 01D0                        | IPC9                            | 31:16     | —     | —     | —     | U1RXIP<2:0>   |       |       | U1RXIS<1:0>   |      |      | —    | —    | —    | U1EIP<2:0>    |      |            | U1EIS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | SPI1TXIP<2:0> |       |       | SPI1TXIS<1:0> |      |      | —    | —    | —    | SPI1RXIP<2:0> |      |            | SPI1RXIS<1:0> |      | 0000 |
| 01E0                        | IPC10                           | 31:16     | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | U1TXIP<2:0>   |      |            | U1TXIS<1:0>   |      | 0000 |
| 01F0                        | IPC11                           | 31:16     | —     | —     | —     | CNDIP<2:0>    |       |       | CNDIS<1:0>    |      |      | —    | —    | —    | CNCIP<2:0>    |      |            | CNCIS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | CNBIP<2:0>    |       |       | CNBIS<1:0>    |      |      | —    | —    | —    | CNAIP<2:0>    |      |            | CNAIS<1:0>    |      | 0000 |
| 0200                        | IPC12                           | 31:16     | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | CNFIP<2:0>    |       |       | CNFIS<1:0>    |      |      | —    | —    | —    | CNEIP<2:0>    |      |            | CNEIS<1:0>    |      | 0000 |
| 0210                        | IPC13                           | 31:16     | —     | —     | —     | SPI2TXIP<2:0> |       |       | SPI2TXIS<1:0> |      |      | —    | —    | —    | SPI2RXIP<2:0> |      |            | SPI2RXIS<1:0> |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | SPI2EIP<2:0>  |       |       | SPI2EIS<1:0>  |      |      | —    | —    | —    | —             | —    | —          | —             | —    | 0000 |
| 0220                        | IPC14                           | 31:16     | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | U2RXIP<2:0>   |       |       | U2RXIS<1:0>   |      |      | —    | —    | —    | U2EIP<2:0>    |      |            | U2EIS<1:0>    |      | 0000 |
| 0230                        | IPC15                           | 31:16     | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | —    | 0000 |
| 0240                        | IPC16                           | 31:16     | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | —    | 0000 |
| 0250                        | IPC17                           | 31:16     | —     | —     | —     | CTMUIP<2:0>   |       |       | CTMUIS<1:0>   |      |      | —    | —    | —    | —             | —    | —          | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | —             | —    | —          | —             | —    | 0000 |
| 0260                        | IPC18                           | 31:16     | —     | —     | —     | DMA3IP<2:0>   |       |       | DMA3IS<1:0>   |      |      | —    | —    | —    | DMA2IP<2:0>   |      |            | DMA2IS<1:0>   |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | DMA1IP<2:0>   |       |       | DMA1IS<1:0>   |      |      | —    | —    | —    | DMA0IP<2:0>   |      |            | DMA0IS<1:0>   |      | 0000 |
| 0270                        | IPC19                           | 31:16     | —     | —     | —     | OC6IP<2:0>    |       |       | OC6IS<1:0>    |      |      | —    | —    | —    | IC6IP<2:0>    |      |            | IC6IS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | IC6EIP<2:0>   |       |       | IC6EIS<1:0>   |      |      | —    | —    | —    | T6IP<2:0>     |      |            | T6IS<1:0>     |      | 0000 |
| 0280                        | IPC20                           | 31:16     | —     | —     | —     | OC7IP<2:0>    |       |       | OC7IS<1:0>    |      |      | —    | —    | —    | IC7IP<2:0>    |      |            | IC7IS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | IC7EIP<2:0>   |       |       | IC7EIS<1:0>   |      |      | —    | —    | —    | T7IP<2:0>     |      |            | T7IS<1:0>     |      | 0000 |
| 0290                        | IPC21                           | 31:16     | —     | —     | —     | OC8IP<2:0>    |       |       | OC8IS<1:0>    |      |      | —    | —    | —    | IC8IP<2:0>    |      |            | IC8IS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | IC8EIP<2:0>   |       |       | IC8EIS<1:0>   |      |      | —    | —    | —    | T8IP<2:0>     |      |            | T8IS<1:0>     |      | 0000 |
| 02A0                        | IPC22                           | 31:16     | —     | —     | —     | OC9IP<2:0>    |       |       | OC9IS<1:0>    |      |      | —    | —    | —    | IC9IP<2:0>    |      |            | IC9IS<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | IC9EIP<2:0>   |       |       | IC9EIS<1:0>   |      |      | —    | —    | —    | T9IP<2:0>     |      |            | T9IS<1:0>     |      | 0000 |
| 02B0                        | IPC23                           | 31:16     | —     | —     | —     | AD1DC2IP<2:0> |       |       | AD1DC2IS<1:0> |      |      | —    | —    | —    | AD1DC1IP<2:0> |      |            | AD1DC1IS<1:0> |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —    | AD1IP<2:0>    |      |            | AD1IS<1:0>    |      | 0000 |

**Legend:** × = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See **10.2 "CLR, SET, and INV Registers"** for more information.
- 2: This bit is not available on 64-pin devices.
- 3: This bit is not available on devices without a CAN module.
- 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |                |       |       |               |      |      |      |      |      |                |      | All Resets    |                |      |               |      |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|----------------|-------|-------|---------------|------|------|------|------|------|----------------|------|---------------|----------------|------|---------------|------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12          | 27/11 | 26/10 | 25/9          | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3           | 18/2 |               | 17/1           | 16/0 |               |      |      |
| 02C0                        | IPC24                           | 31:16     | —     | —     | —     | AD1DF4IP<2:0>  |       |       | AD1DF4IS<1:0> |      |      | —    | —    | —    | AD1DF3IP<2:0>  |      |               | AD1DF3IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1DF2IP<2:0>  |       |       | AD1DF2IS<1:0> |      |      | —    | —    | —    | AD1DF1IP<2:0>  |      |               | AD1DF1IS<1:0>  |      | 0000          |      |      |
| 02D0                        | IPC25                           | 31:16     | —     | —     | —     | AD1RSISIP<2:0> |       |       | AD1RSIS<1:0>  |      |      | —    | —    | —    | AD1ARIP<2:0>   |      |               | AD1ARIS<1:0>   |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1EOSIP<2:0>  |       |       | AD1EOSIS<1:0> |      |      | —    | —    | —    | AD1F1IP<2:0>   |      |               | AD1F1IS<1:0>   |      | 0000          |      |      |
| 02E0                        | IPC26                           | 31:16     | —     | —     | —     | AD1D01IP<2:0>  |       |       | AD1D01IS<1:0> |      |      | —    | —    | —    | AD1D00IP<2:0>  |      |               | AD1D00IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1G1IP<2:0>   |       |       | AD1G1IS<1:0>  |      |      | —    | —    | —    | AD1FCBTIP<2:0> |      |               | AD1FCBTIS<1:0> |      | 0000          |      |      |
| 02F0                        | IPC27                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | AD1D04IP<2:0> |                |      | AD1D04IS<1:0> |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | AD1D03IP<2:0>  |       |       | AD1D03IS<1:0> |      |      | —    | —    | —    | AD1D02IP<2:0>  |      |               | AD1D02IS<1:0>  |      | 0000          |      |      |
| 0300                        | IPC28                           | 31:16     | —     | —     | —     | AD1D09IP<2:0>  |       |       | AD1D09IS<1:0> |      |      | —    | —    | —    | AD1D08IP<2:0>  |      |               | AD1D08IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D07IP<2:0>  |       |       | AD1D07IS<1:0> |      |      | —    | —    | —    | AD1D06IP<2:0>  |      |               | AD1D06IS<1:0>  |      | 0000          |      |      |
| 0310                        | IPC29                           | 31:16     | —     | —     | —     | AD1D13IP<2:0>  |       |       | AD1D13IS<1:0> |      |      | —    | —    | —    | AD1D12IP<2:0>  |      |               | AD1D12IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D11IP<2:0>  |       |       | AD1D11IS<1:0> |      |      | —    | —    | —    | AD1D10IP<2:0>  |      |               | AD1D10IS<1:0>  |      | 0000          |      |      |
| 0320                        | IPC30                           | 31:16     | —     | —     | —     | AD1D17IP<2:0>  |       |       | AD1D17IS<1:0> |      |      | —    | —    | —    | AD1D16IP<2:0>  |      |               | AD1D16IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D15IP<2:0>  |       |       | AD1D15IS<1:0> |      |      | —    | —    | —    | AD1D14IP<2:0>  |      |               | AD1D14IS<1:0>  |      | 0000          |      |      |
| 0330                        | IPC31                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D19IP<2:0>  |       |       | AD1D19IS<1:0> |      |      | —    | —    | —    | AD1D18IP<2:0>  |      |               | AD1D18IS<1:0>  |      | 0000          |      |      |
| 0340                        | IPC32                           | 31:16     | —     | —     | —     | AD1D25IP<2:0>  |       |       | AD1D25IS<1:0> |      |      | —    | —    | —    | AD1D24IP<2:0>  |      |               | AD1D24IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
| 0350                        | IPC33                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D27IP<2:0>  |       |       | AD1D27IS<1:0> |      |      | —    | —    | —    | AD1D26IP<2:0>  |      |               | AD1D26IS<1:0>  |      | 0000          |      |      |
| 0360                        | IPC34                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
| 0370                        | IPC35                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
| 0380                        | IPC36                           | 31:16     | —     | —     | —     | AD1D41IP<2:0>  |       |       | AD1D41IS<1:0> |      |      | —    | —    | —    | AD1D40IP<2:0>  |      |               | AD1D40IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
| 0390                        | IPC37                           | 31:16     | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —              | —     | —     | —             | —    | —    | —    | —    | —    | —              | —    | —             | —              | —    | —             | 0000 |      |
| 03A0                        | IPC38                           | 31:16     | —     | —     | —     | AD1D49IP<2:0>  |       |       | AD1D49IS<1:0> |      |      | —    | —    | —    | AD1D48IP<2:0>  |      |               | AD1D48IS<1:0>  |      | 0000          |      |      |
|                             |                                 | 15:0      | —     | —     | —     | AD1D47IP<2:0>  |       |       | AD1D47IS<1:0> |      |      | —    | —    | —    | AD1D46IP<2:0>  |      |               | AD1D46IS<1:0>  |      | 0000          |      |      |

**Legend:** × = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See **10.2 “CLR, SET, and INV Registers”** for more information.
- 2: This bit is not available on 64-pin devices.
- 3: This bit is not available on devices without a CAN module.
- 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |                            |       |       |                            |      |      |      |      |      |                |               | All Resets   |                 |      |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|----------------------------|-------|-------|----------------------------|------|------|------|------|------|----------------|---------------|--------------|-----------------|------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12                      | 27/11 | 26/10 | 25/9                       | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3           | 18/2          |              | 17/1            | 16/0 |      |
| 03B0                        | IPC39                           | 31:16     | —     | —     | —     | AD1D53IP<2:0>              |       |       | AD1D53IS<1:0>              |      |      | —    | —    | —    | AD1D52IP<2:0>  |               |              | AD1D52IS<1:0>   |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | AD1D50IP<2:0> |              | AD1D50IS<1:0>   |      | 0000 |
| 03C0                        | IPC40                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | CMP5IP<2:0>   |              | CMP5IS<1:0>     |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | CMP4IP<2:0>                |       |       | CMP4IS<1:0>                |      |      | —    | —    | —    | CMP3IP<2:0>    |               | CMP3IS<1:0>  |                 | 0000 |      |
| 03D0                        | IPC41                           | 31:16     | —     | —     | —     | CAN1IP<2:0> <sup>(3)</sup> |       |       | CAN1IS<1:0> <sup>(3)</sup> |      |      | —    | —    | —    | —              | —             | —            | —               | —    | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | —    | 0000 |
| 03E0                        | IPC42                           | 31:16     | —     | —     | —     | PWMPEVTIP<2:0>             |       |       | PWMSEVTIP<1:0>             |      |      | —    | —    | —    | QE12IP<2:0>    |               |              | QE12ISIP<1:0>   |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | QE11IP<2:0>                |       |       | QE11SIP<1:0>               |      |      | —    | —    | —    | —              | —             | —            | —               | —    | 0000 |
| 03F0                        | IPC43                           | 31:16     | —     | —     | —     | PWM3IP<2:0>                |       |       | PWM3SIP<1:0>               |      |      | —    | —    | —    | PWM2IP<2:0>    |               |              | PWM2SIP<1:0>    |      | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | PWM1IP<2:0>                |       |       | PWM1SIP<1:0>               |      |      | —    | —    | —    | PWMSEVTIP<2:0> |               |              | PWMSEVTSIP<1:0> |      | 0000 |
| 0400                        | IPC44                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | PWM6IP<2:0>    |               | PWM6SIP<1:0> |                 | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | PWM5IP<2:0>                |       |       | PWM5SIP<1:0>               |      |      | —    | —    | —    | PWM4IP<2:0>    |               | PWM4SIP<1:0> |                 | 0000 |      |
| 0410                        | IPC45                           | 31:16     | —     | —     | —     | DMA5IP<2:0>                |       |       | DMA5IS<1:0>                |      |      | —    | —    | —    | DMA4IP<2:0>    |               | DMA4IS<1:0>  |                 | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0420                        | IPC46                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | DMA7IP<2:0>                |       |       | DMA7IS<1:0>                |      |      | —    | —    | —    | DMA6IP<2:0>    |               | DMA6IS<1:0>  |                 | 0000 |      |
| 0430                        | IPC47                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | QE13IP<2:0>                |       |       | QE13SIP<1:0>               |      |      | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0440                        | IPC48                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0450                        | IPC49                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0460                        | IPC50                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0470                        | IPC51                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0480                        | IPC52                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
| 0490                        | IPC53                           | 31:16     | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |
|                             |                                 | 15:0      | —     | —     | —     | —                          | —     | —     | —                          | —    | —    | —    | —    | —    | —              | —             | —            | —               | 0000 |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
- 2: This bit is not available on 64-pin devices.
- 3: This bit is not available on devices without a CAN module.
- 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.



TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |               |       |       |               |      |      |      |      |               |      |               |             | All Resets |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|---------------|-------|-------|---------------|------|------|------|------|---------------|------|---------------|-------------|------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12         | 27/11 | 26/10 | 25/9          | 24/8 | 23/7 | 22/6 | 21/5 | 20/4          | 19/3 | 18/2          | 17/1        |            | 16/0 |
| 04A0                        | IPC54                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 04B0                        | IPC55                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 04C0                        | IPC56                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 04D0                        | IPC57                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | SBIP<2:0>     |      | SBIS<1:0>     |             | 0000       |      |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 04F0                        | IPC59                           | 31:16     | —          | —     | —     | PWM8IP<2:0>   |       |       | PWM8SIP<1:0>  |      |      | —    | —    | PWM7IP<2:0>   |      | PWM7SIP<1:0>  |             | 0000       |      |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 0500                        | IPC60                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | PWM9IP<2:0>   |      | PWM9SIP<1:0>  |             | 0000       |      |
| 0510                        | IPC61                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | AD1DC4IP<2:0> |      | AD1DC4IS<1:0> |             | 0000       |      |
|                             |                                 | 15:0      | —          | —     | —     | AD1DC3IP<2:0> |       |       | AD1DC3IS<1:0> |      |      | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 0530                        | IPC63                           | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | CPCIP<2:0>    |      | CPCIS<1:0>    |             | 0000       |      |
|                             |                                 | 15:0      | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | —           | —          | 0000 |
| 0540                        | OFF000                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 0544                        | OFF001                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 0548                        | OFF002                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 054C                        | OFF003                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 0550                        | OFF004                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 0554                        | OFF005                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |
| 0558                        | OFF006                          | 31:16     | —          | —     | —     | —             | —     | —     | —             | —    | —    | —    | —    | —             | —    | —             | VOFF<17:16> |            | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |               |       |       |               |      |      |      |      |               |      |               |             | —          | 0000 |

**Legend:** × = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
- 2: This bit is not available on 64-pin devices.
- 3: This bit is not available on devices without a CAN module.
- 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 055C                        | OFF007                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0560                        | OFF008                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0564                        | OFF009                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0568                        | OFF010                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 056C                        | OFF011                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0570                        | OFF012                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0574                        | OFF013                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0578                        | OFF014                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 057C                        | OFF015                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0580                        | OFF016                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0584                        | OFF017                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0588                        | OFF018                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 058C                        | OFF019                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0590                        | OFF020                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0594                        | OFF021                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 0598                        | OFF022                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 059C                        | OFF023                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05A0                        | OFF024                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05A4                        | OFF025                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05A8                        | OFF026                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05AC                        | OFF027                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05B8                        | OFF030                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05BC                        | OFF031                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05C0                        | OFF032                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05C4                        | OFF033                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05CC                        | OFF035                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05D0                        | OFF036                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05D4                        | OFF037                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05D8                        | OFF038                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05DC                        | OFF039                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
- 2: This bit is not available on 64-pin devices.
- 3: This bit is not available on devices without a CAN module.
- 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 05E0                        | OFF040                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05E4                        | OFF041                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05E8                        | OFF042                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05EC                        | OFF043                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05F0                        | OFF044                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05F4                        | OFF045                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05F8                        | OFF046                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 05FC                        | OFF047                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0600                        | OFF048                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0604                        | OFF049                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0608                        | OFF050                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0614                        | OFF053                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0618                        | OFF054                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 061C                        | OFF055                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0620                        | OFF056                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 0624                        | OFF057                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0628                        | OFF058                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 062C                        | OFF059                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0630                        | OFF060                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0634                        | OFF061                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 065C                        | OFF071                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0660                        | OFF072                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0664                        | OFF073                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0668                        | OFF074                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 066C                        | OFF075                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0670                        | OFF076                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0674                        | OFF077                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0678                        | OFF078                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 067C                        | OFF079                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0680                        | OFF080                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 0684                        | OFF081                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0688                        | OFF082                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 068C                        | OFF083                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0690                        | OFF084                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0694                        | OFF085                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0698                        | OFF086                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 069C                        | OFF087                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06A0                        | OFF088                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06A4                        | OFF089                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06A8                        | OFF090                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06AC                        | OFF091                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06B0                        | OFF092                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06B8                        | OFF094                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06BC                        | OFF095                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 06C0                        | OFF096                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets  |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |             | 17/1 |
| 06C4                        | OFF097                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06C8                        | OFF098                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06CC                        | OFF099                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06D0                        | OFF100                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06D4                        | OFF101                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06D8                        | OFF102                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06DC                        | OFF103                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06E0                        | OFF104                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06E4                        | OFF105                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06E8                        | OFF106                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06EC                        | OFF107                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06F0                        | OFF108                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06F4                        | OFF109                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06F8                        | OFF110                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |
| 06FC                        | OFF111                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —           | 0000 |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 0700                        | OFF112                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0704                        | OFF113                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0708                        | OFF114                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 070C                        | OFF115                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0710                        | OFF116                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0714                        | OFF117                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0718                        | OFF118                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 071C                        | OFF119                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0720                        | OFF120                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0724                        | OFF121                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0728                        | OFF122                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 072C                        | OFF123                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0730                        | OFF124                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0734                        | OFF125                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0748                        | OFF130                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.



TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 074C                        | OFF131                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0750                        | OFF132                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0754                        | OFF133                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0788                        | OFF146                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 078C                        | OFF147                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07A0                        | OFF152                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07A4                        | OFF153                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07A8                        | OFF154                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07AC                        | OFF155                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07B0                        | OFF156                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07B8                        | OFF158                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07BC                        | OFF159                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07C0                        | OFF160                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07C4                        | OFF161                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07C8                        | OFF162                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See 10.2 “CLR, SET, and INV Registers” for more information.
- Note 2: This bit is not available on 64-pin devices.
- Note 3: This bit is not available on devices without a CAN module.
- Note 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

**TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 07DC                        | OFF167                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07E4                        | OFF169                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07E8                        | OFF170                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07EC                        | OFF171                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07F0                        | OFF172                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07F4                        | OFF173                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07F8                        | OFF174                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 07FC                        | OFF175                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0800                        | OFF176                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0804                        | OFF177                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0808                        | OFF178                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0818                        | OFF182                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 081C                        | OFF183                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0820                        | OFF184                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0824                        | OFF185                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

TABLE 6-4: INTERRUPT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |             |      |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|-------------|------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1        | 16/0 |
| 0834                        | OFF189                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 08D8                        | OFF230                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 08F8                        | OFF238                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 08FC                        | OFF239                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0900                        | OFF240                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0914                        | OFF245                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0918                        | OFF246                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 091C                        | OFF247                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0924                        | OFF249                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0928                        | OFF250                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0938                        | OFF254                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |
| 0938C                       | OFF255                          | 31:16     | —          | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | VOFF<17:16> | 0000 |
|                             |                                 | 15:0      | VOFF<15:1> |       |       |       |       |       |      |      |      |      |      |      |      |      | —          | 0000        |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
  - 2: This bit is not available on 64-pin devices.
  - 3: This bit is not available on devices without a CAN module.
  - 4: The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition is occurred. The IFSx bits are persistent, hence they must be cleared if they are set by user software after an IFSx user bit interrogation.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 6-1: INTCON: INTERRUPT CONTROL REGISTER**

| Bit Range   | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| NMIKEY<7:0> |                |                |                |                |                |                |               |               |
| 23:16       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
| —           |                |                |                |                |                |                |               |               |
| 15:8        | U-0            | U-0            | U-0            | R/W-0          | U-0            | R/W-0          | R/W-0         | R/W-0         |
| —           |                |                |                |                |                |                |               |               |
|             |                |                |                | MVEC           | TPC<2:0>       |                |               |               |
| 7:0         | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| —           |                |                |                |                |                |                |               |               |
|             |                |                |                | INT4EP         | INT3EP         | INT2EP         | INT1EP        | INT0EP        |

**Legend:**

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

- bit 31-24 **NMIKEY<7:0>**: Software Generated NMI Key Register bits  
 Software NMI event when the correct key (4Eh) is written.  
 Software NMI event not generated when any other value (not the key) is written.
- bit 23-13 **Unimplemented**: Read as '0'
- bit 12 **MVEC**: Multi Vector Configuration bit  
 1 = Interrupt controller configured for multi vectored mode  
 0 = Interrupt controller configured for single vectored mode
- bit 11 **Unimplemented**: Read as '0'
- bit 10-8 **TPC<2:0>**: Interrupt Proximity Timer Control bits  
 111 =Interrupts of group priority 7 or lower start the Interrupt Proximity timer  
 110 =Interrupts of group priority 6 or lower start the Interrupt Proximity timer  
 101 =Interrupts of group priority 5 or lower start the Interrupt Proximity timer  
 100 =Interrupts of group priority 4 or lower start the Interrupt Proximity timer  
 011 =Interrupts of group priority 3 or lower start the Interrupt Proximity timer  
 010 =Interrupts of group priority 2 or lower start the Interrupt Proximity timer  
 001 =Interrupts of group priority 1 start the Interrupt Proximity timer  
 000 =Disables Interrupt Proximity timer
- bit 7-5 **Unimplemented**: Read as '0'
- bit 4 **INT4EP**: External Interrupt 4 Edge Polarity Control bit  
 1 = Rising edge  
 0 = Falling edge
- bit 3 **INT3EP**: External Interrupt 3 Edge Polarity Control bit  
 1 = Rising edge  
 0 = Falling edge
- bit 2 **INT2EP**: External Interrupt 2 Edge Polarity Control bit  
 1 = Rising edge  
 0 = Falling edge
- bit 1 **INT1EP**: External Interrupt 1 Edge Polarity Control bit  
 1 = Rising edge  
 0 = Falling edge
- bit 0 **INT0EP**: External Interrupt 0 Edge Polarity Control bit  
 1 = Rising edge  
 0 = Falling edge

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 6-2: PRIS: PRIORITY SHADOW SELECT REGISTER

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3             | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|----------------|----------------|----------------------------|----------------|---------------|---------------|
| 31:24     | R/W-0                      | R/W-0          | R/W-0          | R/W-0          | R/W-0                      | R/W-0          | R/W-0         | R/W-0         |
|           | PRI7SS<3:0> <sup>(1)</sup> |                |                |                | PRI6SS<3:0> <sup>(1)</sup> |                |               |               |
| 23:16     | R/W-0                      | R/W-0          | R/W-0          | R/W-0          | R/W-0                      | R/W-0          | R/W-0         | R/W-0         |
|           | PRI5SS<3:0> <sup>(1)</sup> |                |                |                | PRI4SS<3:0> <sup>(1)</sup> |                |               |               |
| 15:8      | R/W-0                      | R/W-0          | R/W-0          | R/W-0          | R/W-0                      | R/W-0          | R/W-0         | R/W-0         |
|           | PRI3SS<3:0>                |                |                |                | PRI2SS<3:0> <sup>(1)</sup> |                |               |               |
| 7:0       | R/W-0                      | R/W-0          | R/W-0          | R/W-0          | U-0                        | U-0            | U-0           | R/W-0         |
|           | PRI1SS<3:0> <sup>(1)</sup> |                |                |                | —                          | —              | —             | SS0           |

### Legend:

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **PRI7SS<3:0>**: Interrupt with Priority Level 7 Shadow Set bits<sup>(1)</sup>

1xxx = Reserved (by default, an interrupt with a priority level of 7 uses Shadow Set 0)

•  
•

0111 = Interrupt with a priority level of 7 uses Shadow Set 7

0110 = Interrupt with a priority level of 7 uses Shadow Set 6

•  
•

0001 = Interrupt with a priority level of 7 uses Shadow Set 1

0000 = Interrupt with a priority level of 7 uses Shadow Set 0 (default)

bit 27-24 **PRI6SS<3:0>**: Interrupt with Priority Level 6 Shadow Set bits<sup>(1)</sup>

1xxx = Reserved (by default, an interrupt with a priority level of 6 uses Shadow Set 0)

•  
•

0111 = Interrupt with a priority level of 6 uses Shadow Set 7

0110 = Interrupt with a priority level of 6 uses Shadow Set 6

•  
•

0001 = Interrupt with a priority level of 6 uses Shadow Set 1

0000 = Interrupt with a priority level of 6 uses Shadow Set 0 (default)

bit 23-20 **PRI5SS<3:0>**: Interrupt with Priority Level 5 Shadow Set bits<sup>(1)</sup>

1xxx = Reserved (by default, an interrupt with a priority level of 5 uses Shadow Set 0)

•  
•

0111 = Interrupt with a priority level of 5 uses Shadow Set 7

0110 = Interrupt with a priority level of 5 uses Shadow Set 6

•  
•

0001 = Interrupt with a priority level of 5 uses Shadow Set 1

0000 = Interrupt with a priority level of 5 uses Shadow Set 0 (default)

**Note 1:** These bits are ignored if the MVEC bit (INTCON<12>) = 0.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 6-2: PRIS: PRIORITY SHADOW SELECT REGISTER (CONTINUED)

- bit 19-16 **PRI4SS<3:0>**: Interrupt with Priority Level 4 Shadow Set bits<sup>(1)</sup>
- 1xxx = Reserved (by default, an interrupt with a priority level of 4 uses Shadow Set 0)
  - .
  - .
  - 0111 = Interrupt with a priority level of 4 uses Shadow Set 7
  - 0110 = Interrupt with a priority level of 4 uses Shadow Set 6
  - .
  - .
  - 0001 = Interrupt with a priority level of 4 uses Shadow Set 1
  - 0000 = Interrupt with a priority level of 4 uses Shadow Set 0 (default)
- bit 15-12 **PRI3SS<3:0>**: Interrupt with Priority Level 3 Shadow Set bits<sup>(1)</sup>
- 1xxx = Reserved (by default, an interrupt with a priority level of 3 uses Shadow Set 0)
  - .
  - .
  - 0111 = Interrupt with a priority level of 3 uses Shadow Set 7
  - 0110 = Interrupt with a priority level of 3 uses Shadow Set 6
  - .
  - .
  - 0001 = Interrupt with a priority level of 3 uses Shadow Set 1
  - 0000 = Interrupt with a priority level of 3 uses Shadow Set 0 (default)
- bit 11-8 **PRI2SS<3:0>**: Interrupt with Priority Level 2 Shadow Set bits<sup>(1)</sup>
- 1xxx = Reserved (by default, an interrupt with a priority level of 2 uses Shadow Set 0)
  - .
  - .
  - 0111 = Interrupt with a priority level of 2 uses Shadow Set 7
  - 0110 = Interrupt with a priority level of 2 uses Shadow Set 6
  - .
  - .
  - 0001 = Interrupt with a priority level of 2 uses Shadow Set 1
  - 0000 = Interrupt with a priority level of 2 uses Shadow Set 0 (default)
- bit 7-4 **PRI1SS<3:0>**: Interrupt with Priority Level 1 Shadow Set bits<sup>(1)</sup>
- 1xxx = Reserved (by default, an interrupt with a priority level of 1 uses Shadow Set 0)
  - .
  - .
  - 0111 = Interrupt with a priority level of 1 uses Shadow Set 7
  - 0110 = Interrupt with a priority level of 1 uses Shadow Set 6
  - .
  - .
  - 0001 = Interrupt with a priority level of 1 uses Shadow Set 1
  - 0000 = Interrupt with a priority level of 1 uses Shadow Set 0 (default)
- bit 3-1 **Unimplemented**: Read as '0'
- bit 0 **SS0**: Single Vector Shadow Register Set bit
- 1 = Single vector is presented with a shadow set
  - 0 = Single vector is not presented with a shadow set

**Note 1:** These bits are ignored if the MVEC bit (INTCON<12>) = 0.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 6-3: INTSTAT: INTERRUPT STATUS REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | R-0            | R-0           | R-0           |
|           | —              | —              | —              | —              | —              | SRIPL<2:0>     |               |               |
| 7:0       | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | SIRQ<7:0>      |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31-11 **Unimplemented:** Read as '0'
- bit 10-8 **SRIPL<2:0>:** Requested Priority Level bits for Single Vector Mode bits  
 111-000 = The priority level of the latest interrupt presented to the CPU
- bit 7-6 **Unimplemented:** Read as '0'
- bit 7-0 **SIRQ<7:0>:** Last Interrupt Request Serviced Status bits  
 11111111-00000000 = The last interrupt request number serviced by the CPU

**REGISTER 6-4: IPTMR: INTERRUPT PROXIMITY TIMER REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IPTMR<31:24>   |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IPTMR<23:16>   |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IPTMR<15:8>    |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IPTMR<7:0>     |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31-0 **IPTMR<31:0>:** Interrupt Proximity Timer Reload bits  
 Used by the Interrupt Proximity Timer as a reload value when the Interrupt Proximity timer is triggered by an interrupt event.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 6-5: IFSx: INTERRUPT FLAG STATUS REGISTER 'x' ('x' = 0-7)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IFS31          | IFS30          | IFS29          | IFS28          | IFS27          | IFS26          | IFS25         | IFS24         |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IFS23          | IFS22          | IFS21          | IFS20          | IFS19          | IFS18          | IFS17         | IFS16         |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IFS15          | IFS14          | IFS13          | IFS12          | IFS11          | IFS10          | IFS9          | IFS8          |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IFS7           | IFS6           | IFS5           | IFS4           | IFS3           | IFS2           | IFS1          | IFS0          |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **IFS31-IFS0**: Interrupt Flag Status bits  
 1 = Interrupt request has occurred  
 0 = No interrupt request has occurred

**Note:** This register represents a generic definition of the IFSx register. Refer to [Table 6-3](#) for the exact bit definitions.

**REGISTER 6-6: IECx: INTERRUPT ENABLE CONTROL REGISTER 'x' ('x' = 0-7)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IEC31          | IEC30          | IEC29          | IEC28          | IEC27          | IEC26          | IEC25         | IEC24         |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IEC23          | IEC22          | IEC21          | IEC20          | IEC19          | IEC18          | IEC17         | IEC16         |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IEC15          | IEC14          | IEC13          | IEC12          | IEC11          | IEC10          | IEC9          | IEC8          |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | IEC7           | IEC6           | IEC5           | IEC4           | IEC3           | IEC2           | IEC1          | IEC0          |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **IEC31-IEC0**: Interrupt Enable bits  
 1 = Interrupt is enabled  
 0 = Interrupt is disabled

**Note:** This register represents a generic definition of the IECx register. Refer to [Table 6-3](#) for the exact bit definitions.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 6-7: IPCx: INTERRUPT PRIORITY CONTROL REGISTER 'x' ('x' = 0-63)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | IP3<2:0>       |                |                | IS3<1:0>      |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | IP2<2:0>       |                |                | IS2<1:0>      |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | IP1<2:0>       |                |                | IS1<1:0>      |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | IP0<2:0>       |                |                | IS0<1:0>      |               |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-26 **IP3<2:0>:** Interrupt Priority bits

111 = Interrupt priority is 7

•  
•  
•

010 = Interrupt priority is 2

001 = Interrupt priority is 1

000 = Interrupt is disabled

bit 25-24 **IS3<1:0>:** Interrupt Subpriority bits

11 = Interrupt subpriority is 3

10 = Interrupt subpriority is 2

01 = Interrupt subpriority is 1

00 = Interrupt subpriority is 0

bit 23-21 **Unimplemented:** Read as '0'

bit 20-18 **IP2<2:0>:** Interrupt Priority bits

111 = Interrupt priority is 7

•  
•  
•

010 = Interrupt priority is 2

001 = Interrupt priority is 1

000 = Interrupt is disabled

bit 17-16 **IS2<1:0>:** Interrupt Subpriority bits

11 = Interrupt subpriority is 3

10 = Interrupt subpriority is 2

01 = Interrupt subpriority is 1

00 = Interrupt subpriority is 0

bit 15-13 **Unimplemented:** Read as '0'

**Note:** This register represents a generic definition of the IPCx register. Refer to [Table 6-3](#) for the exact bit definitions.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 6-7: IPCx: INTERRUPT PRIORITY CONTROL REGISTER 'x' ('x' = 0-63) (CONTINUED)

bit 12-10 **IP1<2:0>**: Interrupt Priority bits

111 = Interrupt priority is 7

.

.

010 = Interrupt priority is 2

001 = Interrupt priority is 1

000 = Interrupt is disabled

bit 9-8 **IS1<1:0>**: Interrupt Subpriority bits

11 = Interrupt subpriority is 3

10 = Interrupt subpriority is 2

01 = Interrupt subpriority is 1

00 = Interrupt subpriority is 0

bit 7-5 **Unimplemented**: Read as '0'

bit 4-2 **IP0<2:0>**: Interrupt Priority bits

111 = Interrupt priority is 7

.

.

010 = Interrupt priority is 2

001 = Interrupt priority is 1

000 = Interrupt is disabled

bit 1-0 **IS0<1:0>**: Interrupt Subpriority bits

11 = Interrupt subpriority is 3

10 = Interrupt subpriority is 2

01 = Interrupt subpriority is 1

00 = Interrupt subpriority is 0

**Note:** This register represents a generic definition of the IPCx register. Refer to [Table 6-3](#) for the exact bit definitions.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 6-8: OFFx: INTERRUPT VECTOR ADDRESS OFFSET REGISTER (x = 0-190)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | —              | —              | VOFF<17:16>   |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | VOFF<15:8>     |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | U-0           |
|           | VOFF<7:1>      |                |                |                |                |                |               | —             |

**Legend:**

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

- bit 31-16 **Unimplemented:** Read as '0'
- bit 17-1 **VOFF<17:1>:** Interrupt Vector 'x' Address Offset bits
- bit 0 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

## 7.0 OSCILLATOR CONFIGURATION

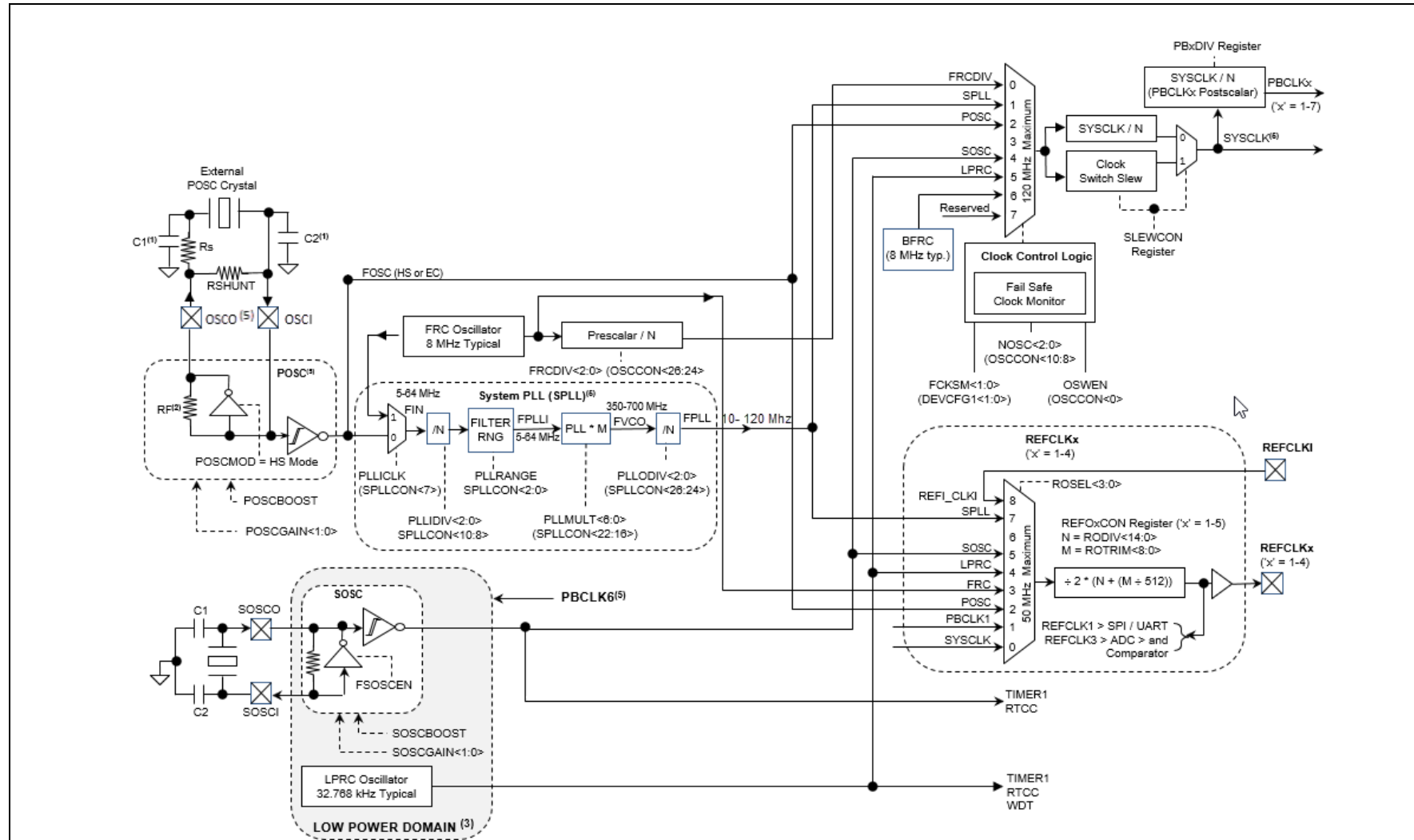
**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 42. “Oscillators with Enhanced PLL”** (DS60001250) in the *“PIC32 Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

The PIC32MK GPG/MCJ with CAN FD Family oscillator system has the following modules and features:

- Five external and internal oscillator options as clock sources
- On-Chip PLL with user-selectable input divider, multiplier and output divider to boost operating frequency on select internal and external oscillator sources
- On-Chip user-selectable divisor postscaler on select oscillator sources
- Software-controllable switching between various clock sources
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shut-down with dedicated Backup FRC (BFRC)
- Flexible reference clock output
- Multiple clock branches for peripherals for better performance flexibility

A block diagram of the oscillator system is provided in [Figure 7-1](#). The clock distribution is shown in [Table 7-1](#).

**FIGURE 7-1: PIC32MK GPG/MCJ WITH CAN FD FAMILY OSCILLATOR DIAGRAM**



- Note**
- 1: Refer to **2.0 "Guidelines for Getting Started with 32-bit MCUs"** for recommended external crystal component values and restrictions.
  - 2: The internal POSC feedback resistor, RF, is typically in the range of 2 to 10 M.
  - 3: The maximum PBCLK6 clock rate to the peripherals in the Low power domain is 30 MHz. This is not the power-up default and must be configured by the user before attempting any access to those peripherals.
  - 4: Refer to **Table 36-16 in 36.0 "Electrical Characteristics"** for PBCLK6 frequency limitations.
  - 5: CLKO on OSCO pin, if enabled in configuration word, available in EC & FRC mode is PBCLK1 / 2.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 7-1: SYSTEM AND PERIPHERAL CLOCK DISTRIBUTION**

| Peripheral          | Clock Source |      |      |      |        |      |                       |        |        |        |          |          |          |          |
|---------------------|--------------|------|------|------|--------|------|-----------------------|--------|--------|--------|----------|----------|----------|----------|
|                     | FRC          | LPRC | SOSC | POSC | SYSCLK | SPLL | PBCLK1 <sup>(1)</sup> | PBCLK2 | PBCLK4 | PBCLK6 | REFCLK01 | REFCLK02 | REFCLK03 | REFCLK04 |
| ADC1-ADC7           |              |      |      |      | X      |      |                       |        |        |        |          |          | X        |          |
| CAN1                |              |      |      |      | X      |      |                       |        |        |        |          |          |          | X        |
| CFG PMD             |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |
| CLKO <sup>(5)</sup> |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |
| Comparator 1-5      |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| CPU                 | X            | X    | X    | X    | X      | X    |                       |        |        |        |          |          |          |          |
| CRU                 |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |
| CTMU                |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| CDAC1-CDAC2         |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| DMA                 |              |      |      |      | X      |      |                       |        |        |        |          |          |          |          |
| DMT                 |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |
| EVIC                |              |      |      |      | X      |      |                       |        |        |        |          |          |          |          |
| Flash               | X            |      |      |      | X      |      | X                     |        |        |        |          |          |          |          |
| Input Capture 1-9   |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| ICD                 |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |
| Output Compare 1-9  |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| Op amp 1-3, 5       |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| PORTA-PORTG         |              |      |      |      |        |      |                       |        | X      |        |          |          |          |          |
| PPS                 |              |      |      |      |        |      | X                     |        |        |        | X        | X        | X        | X        |
| RTCC                |              | X    | X    |      |        |      |                       |        | X      |        |          |          |          |          |
| SPI1-SPI2           |              |      |      |      |        |      |                       | X      |        |        | X        |          |          |          |
| SSX Control         |              |      |      |      | X      |      |                       |        |        |        |          |          |          |          |
| Timer1              |              | X    | X    |      |        |      |                       | X      |        |        |          |          |          |          |
| Timer2-Timer9       |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| UART1-UART2         | X            |      |      |      | X      |      |                       | X      |        |        | X        |          |          |          |
| WDT                 |              | X    |      |      |        |      | X                     |        |        |        |          |          |          |          |
| CLC1-4              |              |      |      |      |        |      |                       | X      |        |        |          |          |          |          |
| HLVD                |              |      |      |      |        |      | X                     |        |        |        |          |          |          |          |

- Note 1:** PBCLK1 is used by system modules and cannot be turned off.  
**Note 2:** SYSCLK is used to fetch data from/to the Flash Controller, while the FRC clock is used for programming.  
**Note 3:** Special Function Register (SFR) access only.  
**Note 4:** Timer1 only.  
**Note 5:** PBCLK1 divided by 2 is available on CLKO function pin on oscillator in EC or FRC mode.

# PIC32MK GPG/MCJ with CAN FD Family

---

## 7.1 Fail-Safe Clock Monitor (FSCM)

The PIC32MK GPG/MCJ with CAN FD Family oscillator system includes a Fail-safe Clock Monitor (FSCM). The FSCM monitors the SYSCLK for continuous operation. If it detects that the SYSCLK has failed, it switches the SYSCLK over to the BFRC oscillator and triggers a NMI. When the NMI is executed, software can attempt to restart the main oscillator or shut down the system.

In Sleep mode both the SYSCLK and the FSCM halt, which prevents FSCM detection.

## 7.2 Oscillator Control Registers

**TABLE 7-2: OSCILLATOR CONFIGURATION REGISTER MAP**

| Virtual Address<br>(BF80_#) | Register Name         | Bit Range | Bits  |       |           |       |       |       |      |      |      |      |      |      |      |      |          | All Resets <sup>(1)</sup> |              |   |      |
|-----------------------------|-----------------------|-----------|-------|-------|-----------|-------|-------|-------|------|------|------|------|------|------|------|------|----------|---------------------------|--------------|---|------|
|                             |                       |           | 31/15 | 30/14 | 29/13     | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1     |                           | 16/0         |   |      |
| 1200                        | OSCCON                | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0xx0 |
|                             |                       | 15:0      | —     | —     | COSC<2:0> |       |       | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1210                        | OSCTUN                | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | TUN<5:0> |                           |              | — | —    |
| 1220                        | SPLLCON               | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0xxx |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | PLLIDIV<2:0> |   | —    |
| 1280                        | REFO1CON              | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | SIDL      | OE    | RSLP  | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1290                        | REFO1TRIM             | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12A0                        | REFO2CON              | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | SIDL      | OE    | RSLP  | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12B0                        | REFO2TRIM             | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12C0                        | REFO3CON              | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | SIDL      | OE    | RSLP  | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12D0                        | REFO3TRIM             | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12E0                        | REFO4CON              | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | SIDL      | OE    | RSLP  | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 12F0                        | REFO4TRIM             | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1300                        | PB1DIV                | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1310                        | PB2DIV                | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1330                        | PB4DIV                | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1350                        | PB6DIV <sup>(2)</sup> | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |
| 1360                        | PB7DIV <sup>(3)</sup> | 31:16     | —     | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | 0000 |
|                             |                       | 15:0      | ON    | —     | —         | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —        | —                         | —            | — | —    |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- Reset values are dependent on the DEVCFGx Configuration bits and the type of reset.
  - Refer to [Table 36-16](#) in [36.0 "Electrical Characteristics"](#) for PBCLK6 frequency limitations.
  - The PB7DIV register is read-only.



**TABLE 7-2: OSCILLATOR CONFIGURATION REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit Range | Bits  |       |       |       |       |             |      |      |      |         |      |         |             |      |         | All Resets <sup>(1)</sup> |        |
|-----------------------------|------------------|-----------|-------|-------|-------|-------|-------|-------------|------|------|------|---------|------|---------|-------------|------|---------|---------------------------|--------|
|                             |                  |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10       | 25/9 | 24/8 | 23/7 | 22/6    | 21/5 | 20/4    | 19/3        | 18/2 | 17/1    |                           | 16/0   |
| 1380                        | SLEWCON          | 31:16     | —     | —     | —     | —     | —     | —           | —    | —    | —    | —       | —    | —       | SYSDIV<3:0> |      |         | 0000                      |        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | SLWDIV<2:0> |      |      | —    | —       | —    | —       | —           | UPEN | DNEN    | BUSY                      | 0000   |
| 1390                        | CLKSTAT          | 31:16     | —     | —     | —     | —     | —     | —           | —    | —    | —    | —       | —    | —       | —           | —    | —       | —                         | 0000   |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —           | —    | —    | —    | SPLLRDY | —    | LPRCRDY | SOSCRDY     | —    | POSCRDY | —                         | FRCRDY |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: Reset values are dependent on the DEVCFGx Configuration bits and the type of reset.
  - 2: Refer to [Table 36-16](#) in [36.0 “Electrical Characteristics”](#) for PBCLK6 frequency limitations.
  - 3: The PB7DIV register is read-only.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 7-1: OSCCON: OSCILLATOR CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0        |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|----------------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0         | R/W-0                |
|           | —              | —              | —              | —              | —              | FRCDIV<2:0>    |               |                      |
| 23:16     | R/W-0          | U-0            | R/W-y          | U-0            | U-0            | U-0            | U-0           | U-0                  |
|           | DRMEN          | —              | SLP2SPD        | —              | —              | —              | —             | —                    |
| 15:8      | U-0            | R-0            | R-0            | R-0            | U-0            | R/W-y          | R/W-y         | R/W-y                |
|           | —              | COSC<2:0>      |                |                | —              | NOSC<2:0>      |               |                      |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0          | R/W-0, HS      | U-0            | R/W-y         | R/W-y                |
|           | CLKLOCK        | —              | —              | SLPEN          | CF             | —              | SOSCEN        | OSWEN <sup>(1)</sup> |

|                   |  |                                    |
|-------------------|--|------------------------------------|
| <b>Legend:</b>    | y = Value set from Configuration bits on POR | HS = Hardware Set                  |
| R = Readable bit  | W = Writable bit                             | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set                             | '0' = Bit is cleared               |
|                   |  | x = Bit is unknown                 |

bit 31-27 **Unimplemented:** Read as '0'

bit 26-24 **FRCDIV<2:0>:** Internal Fast RC (FRC) Oscillator Clock Divider bits

- 111 = FRC divided by 256
- 110 = FRC divided by 64
- 101 = FRC divided by 32
- 100 = FRC divided by 16
- 011 = FRC divided by 8
- 010 = FRC divided by 4
- 001 = FRC divided by 2
- 000 = FRC divided by 1 (default setting)

bit 23 **DRMEN:** Dream Mode Enable bit

- 1 = Dream mode is enabled
- 0 = Dream mode is disabled

bit 22 **Unimplemented:** Read as '0'

bit 21 **SLP2SPD:** Sleep Two-speed Start-up Control bit

- 1 = Use FRC as SYSCLK until the selected clock is ready
- 0 = Use the selected clock directly

bit 20-15 **Unimplemented:** Read as '0'

bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits

- 111 = Reserved
- 110 = Backup Fast RC (BFRC) Oscillator
- 101 = Internal Low-Power RC (LPRC) Oscillator
- 100 = Secondary Oscillator (SOSC)
- 011 = Reserved
- 010 = Primary Oscillator (POSC) (HS or EC)
- 001 = System PLL (SPLL) input clock and divider set by SPLLCN
- 000 = Internal Fast RC (FRC) Oscillator divided by FRCDIV<2:0> bits (FRCDIV) supports FRN / N, where 'N' is 1, 2, 4, 8, 16, 32, 64, and 256

bit 11 **Unimplemented:** Read as '0'

**Note 1:** The reset value for this bit depends on the setting of the IESO bit (DEVCFG1<7>). When IESO = 1, the reset value is '1'. When IESO = 0, the reset value is '0'.

|  |
|--|
| <b>Note:</b> Writes to this register require an unlock sequence. Refer to <b>Section 42. "Oscillators with Enhanced PLL"</b> (DS60001250) in the <i>"PIC32 Family Reference Manual"</i> for details. |
|--|

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 7-1: OSCCON: OSCILLATOR CONTROL REGISTER

bit 10-8 **NOSC<2:0>**: New Oscillator Selection bits

111 = Reserved

110 = Backup Fast RC (BFRC) Oscillator

101 = Internal Low-Power RC (LPRC) Oscillator

100 = Secondary Oscillator (Sosc)

011 = Reserved

010 = Primary Oscillator (Posc) (HS or EC)

001 = System PLL (SPLL) input clock and divider set by SPLLCON

000 = Internal Fast RC (FRC) Oscillator divided by FRCDIV<2:0> bits (FRCDIV) supports FRN/N, where N is 1, 2, 4, 8, 16, 32, 64, and 256

On Reset, these bits are set to the value of the FNOSC<2:0> Configuration bits (DEVCFG1<2:0>).

bit 7 **CLKLOCK**: Clock Selection Lock Enable bit

1 = Clock and PLL selections are locked

0 = Clock and PLL selections are not locked and may be modified

bit 6-5 **Unimplemented**: Read as '0'

bit 4 **SLPEN**: Sleep Mode Enable bit

1 = Device will enter Sleep mode when a WAIT instruction is executed

0 = Device will enter Idle mode when a WAIT instruction is executed

bit 3 **CF**: Clock Fail Detect bit

1 = FSCM has detected a clock failure

0 = No clock failure has been detected

**Note:** On a clock fail event if enabled by the FCKSM<1:0> bits (DEVCFG1<15:14>) = '0b11, this bit and the RNMICON<CF> bit will be set. The user software must clear both the bits inside the CF NMI before attempting to exit the ISR. Software or hardware settings of the CF bit (OSCCON<3>) will cause a CF NMI event and an automatic clock switch to the FRC provided the FCKSM<1:0> = '0b11. Unlike the CF bit (OSCCON<3>), software or hardware settings of the CF bit (RNMICON<17>) will cause a CF NMI event but will not cause a clock switch to the FRC. After a Clock Fail event, a successful user software clock switch if implemented, hardware will automatically clear the CF bit (RNMICON<17>), but not the CF bit (OSCCON<3>). The CF bit (OSCCON<3>) must be cleared by software using the OSCCON register unlock procedure.

bit 2 **Reserved**

bit 1 **SOSCEN**: Secondary Oscillator (Sosc) Enable bit

1 = Enable Secondary Oscillator

0 = Disable Secondary Oscillator

bit 0 **OSWEN**: Oscillator Switch Enable bit<sup>(1)</sup>

1 = Initiate an oscillator switch to selection specified by NOSC<2:0> bits

0 = Oscillator switch is complete

**Note 1:** The reset value for this bit depends on the setting of the IESO bit (DEVCFG1<7>). When IESO = 1, the reset value is '1'. When IESO = 0, the reset value is '0'.

**Note:** Writes to this register require an unlock sequence. Refer to **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the "PIC32 Family Reference Manual" for details.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 7-2: OSCTUN: FRC TUNING REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5          | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|-------------------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0                     | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —                       | —              | —              | —              | —             | —             |
| 23:16     | U-0            | R-0            | U-0                     | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —                       | —              | —              | —              | —             | —             |
| 15:8      | U-0            | R-0            | U-0                     | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —                       | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | R/W-0                   | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | TUN<5:0> <sup>(1)</sup> |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-6 **Unimplemented:** Read as '0'

bit 5-0 **TUN<5:0>**: FRC Oscillator Tuning bits<sup>(1)</sup>

1111111 = +1.453%

- 
- 
- 

1000000 = 0.000% (Nominal Center Frequency, default)

- 
- 
- 

0000000 = -1.500%

**Note 1:** OSCTUN functionality has been provided to help customers compensate for temperature effects on the FRC frequency over a wide range of temperatures. The tuning step size is an approximation, and is neither characterized nor tested.

**Note:** Writes to this register require an unlock sequence. Refer to the **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the *"PIC32 Family Reference Manual"* for details.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 7-3: SPLLCON: SYSTEM PLL CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-y          | R/W-y         | R/W-y         |
|           | —              | —              | —              | —              | —              | PLLODIV<2:0>   |               |               |
| 23:16     | U-0            | R/W-y          | R/W-y          | R/W-y          | R/W-y          | R/W-y          | R/W-y         | R/W-y         |
|           | —              | PLLMULT<6:0>   |                |                |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-y          | R/W-y         | R/W-y         |
|           | —              | PLLIDIV<2:0>   |                |                |                |                |               |               |
| 7:0       | R/W-y          | U-0            | U-0            | U-0            | U-0            | R/W-y          | R/W-y         | R/W-y         |
|           | PLLICK         | —              | —              | —              | —              | PLLRANGE<2:0>  |               |               |

|                   |  |
|-------------------|--|
| <b>Legend:</b>    | y = Value set from Configuration bits on POR                       |
| R = Readable bit  | W = Writable bit      U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown |

bit 31-27 **Unimplemented:** Read as '0'

bit 26-24 **PLLODIV<2:0>:** System PLL Output Clock Divider bits

- 111 = Reserved
- 110 = Reserved
- 101 = PLL Divide by 32
- 100 = PLL Divide by 16
- 011 = PLL Divide by 8
- 010 = PLL Divide by 4
- 001 = PLL Divide by 2
- 000 = Reserved

The default setting is specified by the FPLLODIV<2:0> Configuration bits in the DEVCFG2 register. Refer to [Register 32-5](#) in **32.0 “Special Features”** for information.

bit 23 **Unimplemented:** Read as '0'

bit 22-16 **PLLMULT<6:0>:** System PLL Multiplier bits

- 11111111 = Multiply by 128
- 11111110 = Multiply by 127
- 11111101 = Multiply by 126
- 11111100 = Multiply by 125
- 
- 
- 
- 00000000 = Multiply by 1

The default setting is specified by the FPLLMULT<6:0> Configuration bits in the DEVCFG2 register. Refer to [Register 32-5](#) in **32.0 “Special Features”** for information.

bit 15-11 **Unimplemented:** Read as '0'

**Note 1:** Writes to this register require an unlock sequence. Refer to **Section 42. “Oscillators with Enhanced PLL”** (DS60001250) in the *“PIC32 Family Reference Manual”* for details.

**2:** Writes to this register are not allowed if the SPLLC is selected as a clock source (COSC<2:0> = 001).

**3:** While the PLL is active, and if updating the PLL bits in the OSCCON register at run-time, the user application must remain within the following limits at all times for all nodes in the PLL clock tree. Therefore, the order in which the PLL values may be modified, (i.e., PLLODIV, PLLMULT, PLLIDIV) becomes important. Failure to maintain PLL nodes within min/max ranges may result in unstable PLL and system behavior.

- Output and input to PLLIDIV block (i.e., FPLLI) 5 MHz to 64 MHz (min/max at all times)
- VCO output, (i.e., FVCO) 350 MHz to 700 MHz (min/max at all times)
- Output of PLLODIV, (i.e., FPLL) 10 MHz to 120 MHz (min/max at all times)

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 7-3: SPLLCN: SYSTEM PLL CONTROL REGISTER

bit 10-8 **PLLIDIV<2:0>**: System PLL Input Clock Divider bits

111 = Divide by 8  
110 = Divide by 7  
101 = Divide by 6  
100 = Divide by 5  
011 = Divide by 4  
010 = Divide by 3  
001 = Divide by 2  
000 = Divide by 1

The default setting is specified by the FPLLIDIV<2:0> Configuration bits in the DEVCFG2 register. Refer to [Register 32-5](#) in **32.0 “Special Features”** for information. If the PLLICK is set for FRC, this setting is ignored by the PLL and the divider is set to Divide-by-1.

bit 7 **PLLICK**: System PLL Input Clock Source bit

1 = FRC is selected as the input to the System PLL  
0 = POSC is selected as the input to the System PLL

The POR default is specified by the FPLLICK Configuration bit in the DEVCFG2 register. Refer to [Register 32-5](#) in **32.0 “Special Features”** for information.

bit 6-3 **Unimplemented**: Read as ‘0’

bit 2-0 **PLLRANGE<2:0>**: System PLL Frequency Range Selection bits

111 = Reserved  
110 = 54-64 MHz  
101 = 34-64 MHz  
100 = 21-42 MHz  
011 = 13-26 MHz  
010 = 8-16 MHz  
001 = 5-10 MHz  
000 = Bypass

Use the highest filter range that covers the input freq to the VCO multiplier block that corresponds to the PLLIDIV output freq to minimize PLL system jitter (see [Figure 7-1](#)). For example, Crystal = 20 MHz, PLLIDIV<2:0> = 0b1; therefore, the filter input frequency is equal to 10 MHz, and therefore, PLLRANGE<2:0> = 0b010. The default setting is specified by the FPLLRNG<2:0> Configuration bits in the DEVCFG2 register. Refer to [Register 32-5](#) in **32.0 “Special Features”** for information.

- Note 1:** Writes to this register require an unlock sequence. Refer to **Section 42. “Oscillators with Enhanced PLL”** (DS60001250) in the *“PIC32 Family Reference Manual”* for details.
- 2:** Writes to this register are not allowed if the SPLLCN is selected as a clock source (COSC<2:0> = 001).
- 3:** While the PLL is active, and if updating the PLL bits in the OSCCON register at run-time, the user application must remain within the following limits at all times for all nodes in the PLL clock tree. Therefore, the order in which the PLL values may be modified, (i.e., PLLDIV, PLLMULT, PLLDIV) becomes important. Failure to maintain PLL nodes within min/max ranges may result in unstable PLL and system behavior.
- Output and input to PLLIDIV block (i.e., FPLLI) 5 MHz to 64 MHz (min/max at all times)
  - VCO output, (i.e., FVCO) 350 MHz to 700 MHz (min/max at all times)
  - Output of PLLDIV, (i.e., FPLL) 10 MHz to 120 MHz (min/max at all times)

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 7-4: REFOxCON: REFERENCE OSCILLATOR CONTROL REGISTER ('x' = 1-4)**

| Bit Range | Bit 31/23/15/7            | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3      | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0         |
|-----------|---------------------------|----------------|----------------|----------------|---------------------|----------------|---------------|-----------------------|
| 31:24     | U-0                       | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0          | R/W-0         | R/W-0                 |
|           | RODIV<14:8>               |                |                |                |                     |                |               |                       |
| 23:16     | R/W-0                     | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0          | R/W-0         | R/W-0                 |
|           | RODIV<7:0>                |                |                |                |                     |                |               |                       |
| 15:8      | R/W-0                     | U-0            | R/W-0          | R/W-0          | R/W-0               | U-0            | R/W-0, HC     | R-0, HS, HC           |
|           | ON <sup>(1)</sup>         | —              | SIDL           | OE             | RSLP <sup>(2)</sup> | —              | DIVSWEN       | ACTIVE <sup>(1)</sup> |
| 7:0       | U-0                       | U-0            | U-0            | U-0            | R/W-0               | R/W-0          | R/W-0         | R/W-0                 |
|           | ROSEL<3:0> <sup>(3)</sup> |                |                |                |                     |                |               |                       |

|                   |                       |                                    |
|-------------------|-----------------------|------------------------------------|
| <b>Legend:</b>    | HC = Hardware Cleared | HS = Hardware Set                  |
| R = Readable bit  | W = Writable bit      | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set      | '0' = Bit is cleared               |
|                   |                       | x = Bit is unknown                 |

bit 31 **Unimplemented:** Read as '0'

bit 30-16 **RODIV<14:0>** Reference Clock Divider bits

The value selects the reference clock divider bits (see [Figure 7-1](#) for details). A value of '0' selects no divider.

bit 15 **ON:** Output Enable bit<sup>(1)</sup>

1 = Reference Oscillator Module enabled  
0 = Reference Oscillator Module disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Peripheral Stop in Idle Mode bit

1 = Discontinue module operation when the device enters Idle mode  
0 = Continue module operation in Idle mode

bit 12 **OE:** Reference Clock Output Enable bit

1 = Reference clock is driven out on REFCLKOx pin  
0 = Reference clock is not driven out on REFCLKOx pin

bit 11 **RSLP:** Reference Oscillator Module Run in Sleep bit<sup>(2)</sup>

1 = Reference Oscillator Module output continues to run in Sleep  
0 = Reference Oscillator Module output is disabled in Sleep

bit 10 **Unimplemented:** Read as '0'

bit 9 **DIVSWEN:** Divider Switch Enable bit

1 = Divider switch is in progress  
0 = Divider switch is complete

bit 8 **ACTIVE:** Reference Clock Request Status bit<sup>(1)</sup>

1 = Reference clock request is active  
0 = Reference clock request is not active

bit 7-4 **Unimplemented:** Read as '0'

**Note 1:** Do not write to this register when the ON bit is not equal to the ACTIVE bit.

**2:** This bit is ignored when the ROSEL<3:0> bits = 0000 or 0001.

**3:** The ROSEL<3:0> bits should not be written while the ACTIVE bit is '1', as undefined behavior may result.

**4:** REFOx pin output freq = (ROSEL, Reference Clock Source Freq / (2 \* (RODIV + (ROTRIM / 512)))

**5:** For REFOx resulting frequencies that constitute a fractional result, the circuit will produce an average clock rate, meaning that the circuit will steal cycles from the input clock source such that the REFOx clock output over a period of cycles will average out to the desired frequency. For this reason, unless the resulting clock output is a whole integer value, it is not recommend for use as a clock source for ADC or asynchronous peripherals like CAN or UART do to the resulting jitter and clock rate uncertainty.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 7-4: REFOxCON: REFERENCE OSCILLATOR CONTROL REGISTER ('x' = 1-4)

bit 3-0 **ROSEL<3:0>**: Reference Clock Source Select bits<sup>(3)</sup>

- 1111 = Reserved
- .
- .
- .
- 1001 = Reserved
- 1000 = REFCLKI
- 0111 = SPLL
- 0110 = Reserved
- 0101 = Sosc
- 0100 = LPRC
- 0011 = FRC
- 0010 = Posc
- 0001 = PBCLK1
- 0000 = SYSCLK

- Note 1:** Do not write to this register when the ON bit is not equal to the ACTIVE bit.
- 2:** This bit is ignored when the ROSEL<3:0> bits = 0000 or 0001.
- 3:** The ROSEL<3:0> bits should not be written while the ACTIVE bit is '1', as undefined behavior may result.
- 4:** REFOx pin output freq = (ROSEL, Reference Clock Source Freq / (2 \* (RODIV + (ROTRIM / 512)))
- 5:** For REFOx resulting frequencies that constitute a fractional result, the circuit will produce an average clock rate, meaning that the circuit will steal cycles from the input clock source such that the REFOx clock output over a period of cycles will average out to the desired frequency. For this reason, unless the resulting clock output is a whole integer value, it is not recommend for use as a clock source for ADC or asynchronous peripherals like CAN or UART do to the resulting jitter and clock rate uncertainty.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 7-5: REFOxTRIM: REFERENCE OSCILLATOR TRIM REGISTER ('x' = 1-4)**

| Bit Range   | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ROTRIM<8:1> |                |                |                |                |                |                |               |               |
| 23:16       | R/W-0          | R-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|             | ROTRIM<0>      | —              | —              | —              | —              | —              | —             | —             |
| 15:8        | U-0            | R-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|             | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0         | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|             | —              | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-23 **ROTRIM<8:0>**: Reference Oscillator Trim bits

- 111111111 = 511/512 divisor added to RODIV value
- 111111110 = 510/512 divisor added to RODIV value
- 
- 
- 
- 100000000 = 256/512 divisor added to RODIV value
- 
- 
- 
- 000000010 = 2/512 divisor added to RODIV value
- 000000001 = 1/512 divisor added to RODIV value
- 000000000 = 0 divisor added to RODIV value

bit 22-0 **Unimplemented**: Read as '0'

**Note 1:** While the ON bit (REFOxCON<15>) is '1', writes to this register do not take effect until the DIVSWEN bit is also set to '1'.

**2:** Do not write to this register when the ON bit (REFOxCON<15>) is not equal to the ACTIVE bit (REFOxCON<8>).

**3:** Specified values in this register do not take effect if RODIV<14:0> (REFOxCON<30:16>) = 0.

**4:** REFCLKOx Frequency = ((Selected Source Clock / 2) \* (N + (M / 512)))  
 where, Selected source clock = ROSEL, N = RODIV<14:0>, and M = ROTRIM<8:0>.  
 If the value of REFCLKOx Frequency is not a whole integer value, the output clock will have jitter as it will cause the REFCLKOx circuit to clock cycle steal to produce an average frequency equivalent to the user application's desired frequency. The amount of jitter, (i.e., clock cycle steals), become less as the fractional remainder value becomes closer to a whole number and is greatest at any value plus 0.5.

**5:** REFOx pin output freq = (ROSEL, Reference Clock Source Freq / (2 \* (RODIV + (ROTRIM / 512)))

**6:** For REFOx resulting frequencies that constitute a fractional result, the circuit will produce an average clock rate, meaning that the circuit will steal cycles from the input clock source such that the REFOx clock output over a period of cycles will average out to the desired frequency. For this reason, unless the resulting clock output is a whole integer value, it is not recommend for use as a clock source for ADC or asynchronous peripherals like CAN or UART do to the resulting jitter and clock rate uncertainty.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 7-7: SLEWCON: OSCILLATOR SLEW CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3             | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0                        | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —                          | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | R/W-0                      | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | SYSDIV<3:0> <sup>(1)</sup> |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0                        | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | —                          | SLWDIV<2:0>    |               |               |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0                        | R/W-1          | R/W-0         | R-0, HS, HC   |
|           | —              | —              | —              | —              | —                          | UPEN           | DNEN          | BUSY          |

|                   |                       |                                    |
|-------------------|-----------------------|------------------------------------|
| <b>Legend:</b>    | HC = Hardware Cleared | HS = Hardware Set                  |
| R = Readable bit  | W = Writable bit      | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set      | '0' = Bit is cleared               |
|                   |                       | x = Bit is unknown                 |

bit 31-20 **Unimplemented:** Read as '0'

bit 19-16 **SYSDIV<3:0>:** System Clock Divide Control bits<sup>(1)</sup>

1111 = SYSCLK is divided by 16

1110 = SYSCLK is divided by 15

.

.

.

0010 = SYSCLK is divided by 3

0001 = SYSCLK is divided by 2

0000 = SYSCLK is not divided

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **SLWDIV<2:0>:** Slew Divisor Steps Control bits

These bits control the maximum division steps used when slewing during a frequency change.

111 = Steps are divide by 128, 64, 32, 16, 8, 4, 2, and then no divisor

110 = Steps are divide by 64, 32, 16, 8, 4, 2, and then no divisor

101 = Steps are divide by 32, 16, 8, 4, 2, and then no divisor

100 = Steps are divide by 16, 8, 4, 2, and then no divisor

011 = Steps are divide by 8, 4, 2, and then no divisor

010 = Steps are divide by 4, 2, and then no divisor

001 = Steps are divide by 2, and then no divisor

000 = No divisor is used during slewing

The steps apply in reverse order (i.e., 2, 4, 8, etc.) during a downward frequency change.

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **UPEN:** Upward Slew Enable bit

1 = Slewing enabled for switching to a higher frequency

0 = Slewing disabled for switching to a higher frequency

bit 1 **DNEN:** Downward Slew Enable bit

1 = Slewing enabled for switching to a lower frequency

0 = Slewing disabled for switching to a lower frequency

bit 0 **BUSY:** Clock Switching Slewing Active Status bit

1 = Clock frequency is being actively slewed to the new frequency

0 = Clock switch has reached its final value

**Note 1:** The SYSDIV<3:0> bit settings are ignored if both UPEN and DNEN = 0, and SYSCLK will be divided by 1.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 7-8: CLKSTAT: OSCILLATOR CLOCK STATUS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | R-0            | U-0            | R-0            | R-0            | U-0            | R-0            | U-0           | R-0           |
|           | SPLLRDY        | —              | LPRCRDY        | SOSCRDY        | —              | POSCRDY        | —             | FRCRDY        |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31-8 **Unimplemented:** Read as '0'
- bit 7 **SPLLRDY:** System PLL (SPLL) Ready Status bit  
 1 = SPLL is ready  
 0 = SPLL is not ready
- bit 5 **LPRCRDY:** Low-Power RC (LPRC) Oscillator Ready Status bit  
 1 = LPRC is stable and ready  
 0 = LPRC is disabled or not operating
- bit 4 **SOSCRDY:** Secondary Oscillator (Sosc) Ready Status bit  
 1 = Sosc is stable and ready  
 0 = Sosc is disabled or not operating
- bit 3 **Unimplemented:** Read as '0'
- bit 2 **POSCRDY:** Primary Oscillator (Posc) Ready Status bit  
 1 = Posc is stable and ready  
 0 = Posc is disabled or not operating
- bit 1 **Unimplemented:** Read as '0'
- bit 0 **FRCRDY:** Fast RC (FRC) Oscillator Ready Status bit  
 1 = FRC is stable and ready  
 0 = FRC is disabled for not operating

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 8.0 PREFETCH MODULE

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 4. “Prefetch Cache Module”** (DS60001119), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Prefetch module is a performance enhancing module that is included in the PIC32MK GPG/MCJ with CAN FD Family of devices. When running at high-clock rates, Wait states must be inserted into Program Flash Memory (PFM) read transactions to meet the access time of the PFM. Wait states can be hidden to the core by prefetching and storing instructions in a temporary holding area that the CPU can access quickly. Although the data path to the CPU is 32 bits wide, the data path to the PFM is 128 bits wide. This wide data path provides the same bandwidth to the CPU as a 32-bit path running at four times the frequency.

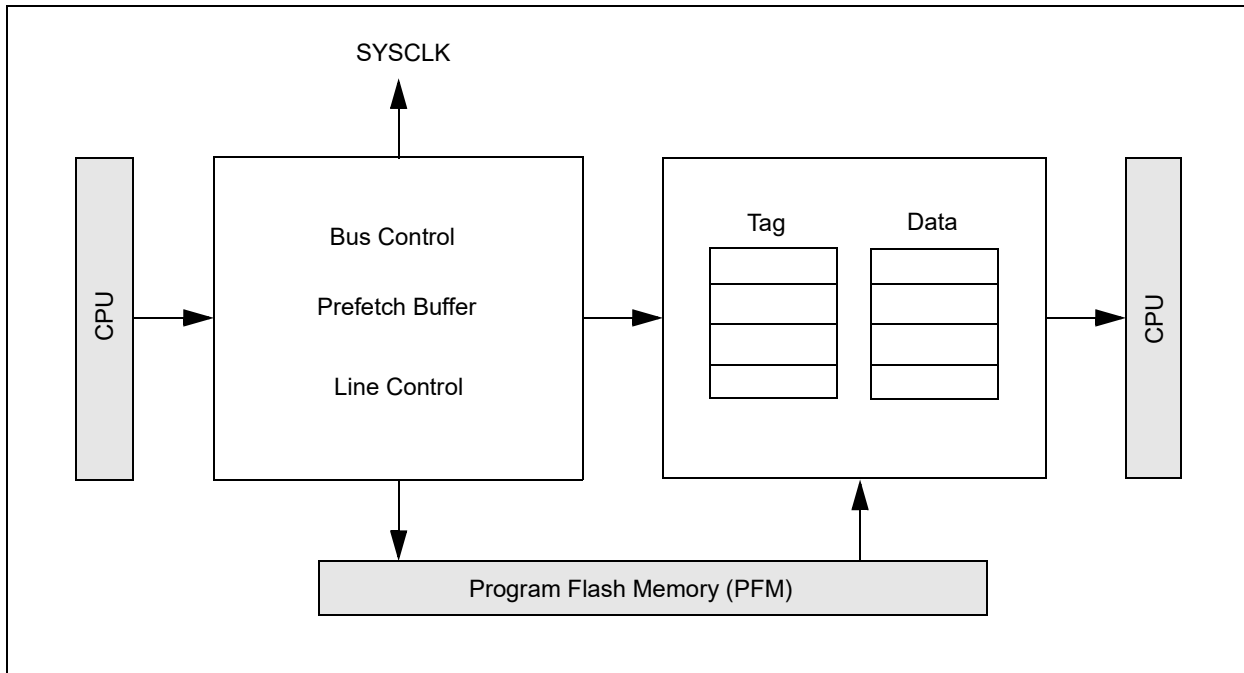
The Prefetch module holds a subset of PFM in temporary holding spaces known as lines. Each line contains a tag and data field. Normally, the lines hold a copy of what is currently in memory to make instructions or data available to the CPU without Flash Wait states.

### 8.1 Prefetch Cache Features

- 36x16 byte fully-associative lines
- 16 lines for CPU instructions
- Four lines for CPU data
- Four lines for peripheral data
- 16-byte parallel memory fetch
- Configurable predictive prefetch

A simplified block diagram of the Prefetch module is shown in [Figure 8-1](#).

**FIGURE 8-1: PREFETCH MODULE BLOCK DIAGRAM**



8.2 Prefetch Control Registers

TABLE 8-1: PREFETCH REGISTER MAP

| Virtual Address (BF80_#) | Register Name <sup>(1)</sup> | Bit Range | Bits          |       |       |           |       |          |        |           |      |           |             |         |      |            |         |         | All Resets |
|--------------------------|------------------------------|-----------|---------------|-------|-------|-----------|-------|----------|--------|-----------|------|-----------|-------------|---------|------|------------|---------|---------|------------|
|                          |                              |           | 31/15         | 30/14 | 29/13 | 28/12     | 27/11 | 26/10    | 25/9   | 24/8      | 23/7 | 22/6      | 21/5        | 20/4    | 19/3 | 18/2       | 17/1    | 16/0    |            |
| 0800                     | CHECON                       | 31:16     | —             | —     | —     | —         | —     | PERCHEEN | DCHEEN | ICHEEN    | —    | PERCHEINV | DCHEINV     | ICHEINV | —    | PERCHECOH  | DCHECOH | ICHECOH | 0700       |
|                          |                              | 15:0      | —             | —     | —     | CHEPERFEN | —     | —        | —      | PFM AWSEN | —    | —         | PREFEN<1:0> |         |      | PFMWS<2:0> |         |         | 0107       |
| 0820                     | CHEHIT                       | 31:16     | CHEHIT<31:16> |       |       |           |       |          |        |           |      |           |             |         |      |            |         |         | 0000       |
|                          |                              | 15:0      | CHEHIT<15:0>  |       |       |           |       |          |        |           |      |           |             |         |      |            |         |         | 0000       |
| 0830                     | CHEMIS                       | 31:16     | CHEMIS<31:16> |       |       |           |       |          |        |           |      |           |             |         |      |            |         |         | 0000       |
|                          |                              | 15:0      | CHEMIS<15:0>  |       |       |           |       |          |        |           |      |           |             |         |      |            |         |         | 0000       |

**Legend:** x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 8-1: CHECON: CACHE MODULE CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6               | Bit 29/21/13/5         | Bit 28/20/12/4         | Bit 27/19/11/3 | Bit 26/18/10/2               | Bit 25/17/9/1          | Bit 24/16/8/0          |
|-----------|----------------|------------------------------|------------------------|------------------------|----------------|------------------------------|------------------------|------------------------|
| 31:24     | U-0            | U-0                          | U-0                    | U-0                    | U-0            | R/W-1                        | R/W-1                  | R/W-1                  |
|           | —              | —                            | —                      | —                      | —              | PERCHEEN                     | DCHEEN                 | ICHEEN                 |
| 23:16     | U-0            | R/W-0                        | R/W-0                  | R/W-0                  | U-0            | R/W-0                        | R/W-0                  | R/W-0                  |
|           | —              | PER<br>CHEINV <sup>(1)</sup> | DCHEINV <sup>(1)</sup> | ICHEINV <sup>(1)</sup> | —              | PER<br>CHECOH <sup>(2)</sup> | DCHECOH <sup>(2)</sup> | ICHECOH <sup>(2)</sup> |
| 15:8      | U-0            | U-0                          | U-0                    | R/W-0                  | U-0            | U-0                          | U-0                    | R/W-1                  |
|           | —              | —                            | —                      | CHE<br>PERFEN          | —              | —                            | —                      | PFM<br>AWSEN           |
| 7:0       | U-0            | U-0                          | R/W-0                  | R/W-0                  | U-0            | R/W-1                        | R/W-1                  | R/W-1                  |
|           | —              | —                            | PREFEN<1:0>            |                        | PFMWS<3:0>     |                              |                        |                        |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-27 **Unimplemented:** Read as '0'

bit 26 **PERCHEEN:** Peripheral Cache Enable bit

- 1 = Peripheral cache is enabled
- 0 = Peripheral cache is disabled

bit 25 **DCHEEN:** Data Cache Enable bit

- 1 = Data cache is enabled
- 0 = Data cache is disabled

bit 24 **ICHEEN:** Instruction Cache Enable bit

- 1 = Instruction cache is enabled
- 0 = Instruction cache is disabled

bit 23 **Unimplemented:** Read as '0'

bit 22 **PERCHEINV:** Peripheral Cache Invalidate bit<sup>(1)</sup>

- 1 = Force invalidate cache/invalidate busy
- 0 = Cache Invalidation follows CHECOH/invalid complete

bit 21 **DCHEINV:** Data Cache Invalidate bit<sup>(1)</sup>

- 1 = Force invalidate cache/invalidate busy
- 0 = Cache Invalidation follows CHECOH/invalid complete

bit 20 **ICHEINV:** Instruction Cache Invalidate bit<sup>(1)</sup>

- 1 = Force invalidate cache/invalidate busy
- 0 = Cache Invalidation follows CHECOH/invalid complete

bit 19 **Unimplemented:** Read as '0'

bit 18 **PERCHECOH:** Peripheral Auto-cache Coherency Control bit<sup>(2)</sup>

- 1 = Automatically invalidate cache on a programming event
- 0 = Do not automatically invalidate cache on a programming event

bit 17 **DCHECOH:** Data Auto-cache Coherency Control bit<sup>(2)</sup>

- 1 = Automatically invalidate cache on a programming event
- 0 = Do not automatically invalidate cache on a programming event

**Note 1:** Hardware automatically clears this bit when cache invalidate completes. Bits may clear at different times.

**Note 2:** The PERCHECOH, DCHECOH, and ICHECOH bits must be stable before initiation of programming.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 8-1: CHECON: CACHE MODULE CONTROL REGISTER (CONTINUED)

- bit 16 **ICHECOH**: Instruction Auto-cache Coherency Control bit<sup>(2)</sup>  
 1 = Automatically invalidate cache on a programming event  
 0 = Do not automatically invalidate cache on a programming event
- bit 15-13 **Unimplemented**: Read as '0'
- bit 12 **CHEPERFEN**: Cache Performance Counters Enable bit  
 1 = Performance counters are enabled  
 0 = Performance counters are disabled
- bit 11-9 **Unimplemented**: Read as '0'
- bit 8 **PFMAWSEN**: PFM Address Wait State Enable bit  
 1 = Add one more Wait State to flash address setup (suggested for higher system clock frequencies)  
 0 = Add no Wait States to the flash address setup (suggested for lower system clock frequencies to achieve higher performance)
- When this bit is set to '1', total Flash wait states are PFMWS plus PFMAWSEN.
- bit 7-6 **Unimplemented**: Read as '0'
- bit 5-4 **PREFEN<1:0>**: Predictive Prefetch Enable bits  
 11 = Disable predictive prefetch  
 10 = Disable predictive prefetch  
 01 = Enable predictive prefetch for CPU instructions only  
 00 = Disable predictive prefetch
- bit 3-0 **PFMWS<3:0>**: PFM Access Time Defined in Terms of SYSCLK Wait States bits  
 0111 = Seven Wait states  
 •  
 •  
 •  
 010 = Two Wait states  
 001 = One Wait state  
 000 = Zero Wait states

| DEVCFG0<9:8>  | Required Flash Wait States<br>PFMWS<3:0> bits | SYSCLK (MHz)               |
|---|---|----------------------------|
| FECCCON<1:0> bits<br>(DEVCFG0<9:8>) = 0x1x<br>with ECC disabled | 1 - Wait State                                | 0 < SYSCLK ≤ 116 MHz       |
|   | 2 - Wait State                                | 116 MHz < SYSCLK ≤ 120 MHz |
| FECCCON<1:0> bits<br>(DEVCFG0<9:8>) = 0x0x<br>with ECC enabled  | 1 - Wait State                                | 0 < SYSCLK ≤ 96 MHz        |
|   | 2 - Wait State                                | 96 MHz < SYSCLK ≤ 120 MHz  |

- Note 1:** Hardware automatically clears this bit when cache invalidate completes. Bits may clear at different times.
- Note 2:** The PERCHECOH, DCHECOH, and ICHECOH bits must be stable before initiation of programming.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 8-2: CHEHIT: CACHE HIT STATUS REGISTER**

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEHIT<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEHIT<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEHIT<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEHIT<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

**bit 31-0 CHEHIT<31:0>: Instruction Cache Hit Count bits**

When the CHEPERFEN bit (CHECON<12>) = 1, the CHEHIT<31:0> bits increment each time the processor issues an instruction fetch or load that hits the prefetch cache from a cacheable region. Non-cacheable accesses do not modify this value.

The CHEHIT<31:0> bits are reset on a '0' to '1' transition of the CHEPERFEN bit.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 8-3: CHEMIS: CACHE MISS STATUS REGISTER

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEMIS<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEMIS<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEMIS<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| CHEMIS<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-0 **CHEMIS<31:0>**: Instruction Cache Miss Count bits

When the CHEPERFEN bit (CHECON<12>) = 1, the CHEMIS<31:0> bits increment each time the processor issues an instruction fetch or load that hits the prefetch cache from a cacheable region. Non-cacheable accesses do not modify this value.

The CHEMIS<31:0> bits are reset on a '0' to '1' transition of the CHEPERFEN bit.

# PIC32MK GPG/MCJ with CAN FD Family

## 9.0 DIRECT MEMORY ACCESS (DMA) CONTROLLER

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 31. “Direct Memory Access (DMA) Controller”** (DS60001117), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

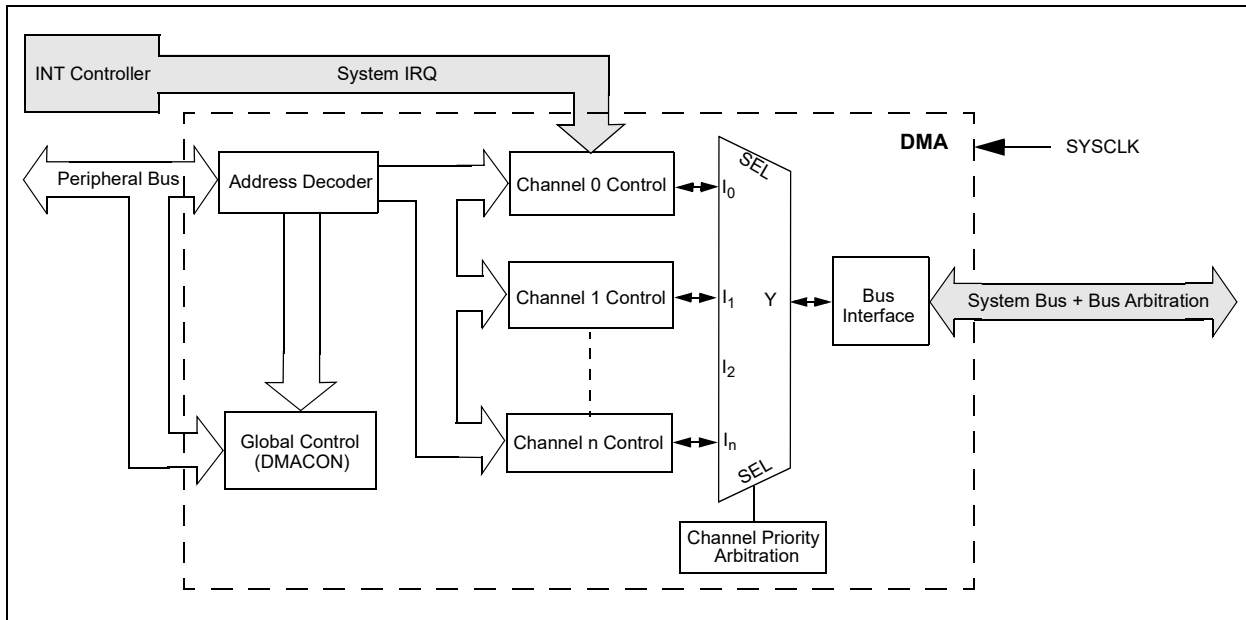
The Direct Memory Access (DMA) Controller is a bus master module useful for data transfers between different devices without CPU intervention. The source and destination of a DMA transfer can be any of the memory mapped modules existent in the device, such as SPI, UART, PMP, etc., or memory itself.

The following are some of the key features of the DMA Controller module:

- Eight identical channels, each featuring:
  - Auto-increment source and destination address registers
  - Source and destination pointers
  - Memory-to-memory and memory-to-peripheral transfers
- Automatic word-size detection:
  - Transfer granularity, down to byte level
  - Bytes need not be word-aligned at source and destination

- Fixed priority channel arbitration
- Flexible DMA channel operating modes:
  - Manual (software) or automatic (interrupt) DMA requests
  - One-Shot or Auto-Repeat Block Transfer modes
  - Channel-to-channel chaining
- Flexible DMA requests:
  - A DMA request can be selected from any of the peripheral interrupt sources
  - Each channel can select any (appropriate) observable interrupt as its DMA request source
  - A DMA transfer abort can be selected from any of the peripheral interrupt sources
  - Up to 2-byte Pattern (data) match transfer termination
- Multiple DMA channel status interrupts:
  - DMA channel block transfer complete
  - Source empty or half empty
  - Destination full or half full
  - DMA transfer aborted due to an external event
  - Invalid DMA address generated
- DMA debug support features:
  - Most recent error address accessed by a DMA channel
  - Most recent DMA channel to transfer data
- CRC Generation module:
  - CRC module can be assigned to any of the available channels
  - CRC module is highly configurable

**FIGURE 9-1: DMA BLOCK DIAGRAM**



## 9.1 DMA Control Registers

**TABLE 9-1: DMA GLOBAL REGISTER MAP**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits          |       |       |         |         |       |      |      |      |      |      |      |      |            |      | All Resets |      |
|-----------------------------|---------------------------------|-----------|---------------|-------|-------|---------|---------|-------|------|------|------|------|------|------|------|------------|------|------------|------|
|                             |                                 |           | 31/15         | 30/14 | 29/13 | 28/12   | 27/11   | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2       | 17/1 |            | 16/0 |
| 1000                        | DMACON                          | 31:16     | —             | —     | —     | —       | —       | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                                 | 15:0      | ON            | —     | —     | SUSPEND | DMABUSY | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
| 1010                        | DMASTAT                         | 31:16     | RDWR          | —     | —     | —       | —       | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000       |      |
|                             |                                 | 15:0      | —             | —     | —     | —       | —       | —     | —    | —    | —    | —    | —    | —    | —    | DMACH<2:0> |      | 0000       |      |
| 1020                        | DMAADDR                         | 31:16     | DMAADDR<31:0> |       |       |         |         |       |      |      |      |      |      |      |      |            |      | 0000       |      |
|                             |                                 | 15:0      | DMAADDR<31:0> |       |       |         |         |       |      |      |      |      |      |      |      |            |      | 0000       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.

**TABLE 9-2: DMA CRC REGISTER MAP**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits           |       |           |           |       |       |      |       |        |        |      |      |            |      |      | All Resets |
|-----------------------------|---------------------------------|-----------|----------------|-------|-----------|-----------|-------|-------|------|-------|--------|--------|------|------|------------|------|------|------------|
|                             |                                 |           | 31/15          | 30/14 | 29/13     | 28/12     | 27/11 | 26/10 | 25/9 | 24/8  | 23/7   | 22/6   | 21/5 | 20/4 | 19/3       | 18/2 | 17/1 |            |
| 1030                        | DCRCCON                         | 31:16     | —              | —     | BYTO<1:0> | WBO       | —     | —     | BITO | —     | —      | —      | —    | —    | —          | —    | —    | 0000       |
|                             |                                 | 15:0      | —              | —     | —         | PLEN<4:0> |       |       |      | CRCEN | CRCAPP | CRCTYP | —    | —    | CRCCH<2:0> |      |      | 0000       |
| 1040                        | DCRCDATA                        | 31:16     | DCRCDATA<31:0> |       |           |           |       |       |      |       |        |        |      |      |            |      |      | 0000       |
|                             |                                 | 15:0      | DCRCDATA<31:0> |       |           |           |       |       |      |       |        |        |      |      |            |      |      | 0000       |
| 1050                        | DCRCXOR                         | 31:16     | DCRCXOR<31:0>  |       |           |           |       |       |      |       |        |        |      |      |            |      |      | 0000       |
|                             |                                 | 15:0      | DCRCXOR<31:0>  |       |           |           |       |       |      |       |        |        |      |      |            |      |      | 0000       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.

**TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP**

| Virtual Address<br>(BF81_#) | Register<br>Name(f) | Bit Range | Bits         |       |          |       |          |       |      |      |        |        |        |        |        |        |        | All Resets |        |      |
|-----------------------------|---------------------|-----------|--------------|-------|----------|-------|----------|-------|------|------|--------|--------|--------|--------|--------|--------|--------|------------|--------|------|
|                             |                     |           | 31/15        | 30/14 | 29/13    | 28/12 | 27/11    | 26/10 | 25/9 | 24/8 | 23/7   | 22/6   | 21/5   | 20/4   | 19/3   | 18/2   | 17/1   |            | 16/0   |      |
| 1060                        | DCH0CON             | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000   |      |
| 1070                        | DCH0ECON            | 31:16     | CHAIRQ<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 00FF       |        |      |
|                             |                     | 15:0      | CHSIRQ<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | FF00       |        |      |
| 1080                        | DCH0INT             | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | CHSDIE | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE | CHERIE     | 0000   |      |
|                             |                     | 15:0      | —            | —     | —        | —     | —        | —     | —    | —    | CHSDIF | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF | CHERIF     | 0000   |      |
| 1090                        | DCH0SSA             | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHSSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10A0                        | DCH0DSA             | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHDSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10B0                        | DCH0SSIZ            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10C0                        | DCH0DSIZ            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10D0                        | DCH0SPTR            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10E0                        | DCH0DPTR            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 10F0                        | DCH0CSIZ            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 1100                        | DCH0CPTR            | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 1110                        | DCH0DAT             | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —      | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                     | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 1120                        | DCH1CON             | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000   |      |
| 1130                        | DCH1ECON            | 31:16     | CHAIRQ<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 00FF       |        |      |
|                             |                     | 15:0      | CHSIRQ<7:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | FF00       |        |      |
| 1140                        | DCH1INT             | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | CHSDIE | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE     | CHERIE | 0000 |
|                             |                     | 15:0      | —            | —     | —        | —     | —        | —     | —    | —    | —      | CHSDIF | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF     | CHERIF | 0000 |
| 1150                        | DCH1SSA             | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHSSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
| 1160                        | DCH1DSA             | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |
|                             |                     | 15:0      | CHDSA<31:0>  |       |          |       |          |       |      |      |        |        |        |        |        |        |        | 0000       |        |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

**TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits         |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | All Resets |
|-----------------------------|---------------------------------|-----------|--------------|-------|----------|-------|----------|-------|------|-------------|--------|--------|--------|--------|--------|--------|------------|------|------------|
|                             |                                 |           | 31/15        | 30/14 | 29/13    | 28/12 | 27/11    | 26/10 | 25/9 | 24/8        | 23/7   | 22/6   | 21/5   | 20/4   | 19/3   | 18/2   | 17/1       | 16/0 |            |
| 1170                        | DCH1SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1180                        | DCH1DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1190                        | DCH1SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 11A0                        | DCH1DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 11B0                        | DCH1CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 11C0                        | DCH1CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 11D0                        | DCH1DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 11E0                        | DCH2CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |            |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | CHCHNS      | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000 |            |
| 11F0                        | DCH2ECON                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | CHAIRQ<7:0> |        |        |        |        |        |        |            | 00FF |            |
|                             |                                 | 15:0      | CHSIRQ<7:0>  |       |          |       |          |       |      | CFORCE      | CABORT | PATEN  | SIRQEN | AIRQEN | —      | —      | —          | FF00 |            |
| 1200                        | DCH2INT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | CHSDIE      | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE | CHERIE     | 0000 |            |
|                             |                                 | 15:0      | —            | —     | —        | —     | —        | —     | —    | CHSDIF      | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF | CHERIF     | 0000 |            |
| 1210                        | DCH2SSA                         | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
|                             |                                 | 15:0      | CHSSA<31:0>  |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1220                        | DCH2DSA                         | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
|                             |                                 | 15:0      | CHDSA<31:0>  |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1230                        | DCH2SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1240                        | DCH2DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1250                        | DCH2SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1260                        | DCH2DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |
| 1270                        | DCH2CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —           | —      | —      | —      | —      | —      | —      | —          | —    | 0000       |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |             |        |        |        |        |        |        |            |      | 0000       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

**TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits         |       |          |       |          |       |      |        |        |        |        |        |        |        |            | All Resets |      |
|-----------------------------|---------------------------------|-----------|--------------|-------|----------|-------|----------|-------|------|--------|--------|--------|--------|--------|--------|--------|------------|------------|------|
|                             |                                 |           | 31/15        | 30/14 | 29/13    | 28/12 | 27/11    | 26/10 | 25/9 | 24/8   | 23/7   | 22/6   | 21/5   | 20/4   | 19/3   | 18/2   | 17/1       |            | 16/0 |
| 1280                        | DCH2CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1290                        | DCH2DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 12A0                        | DCH3CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000       |      |
| 12B0                        | DCH3ECON                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | 00FF       |      |
|                             |                                 | 15:0      | CHAIRQ<7:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | FF00       |      |
| 12C0                        | DCH3INT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIE | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE     | CHERIE     | 0000 |
|                             |                                 | 15:0      | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIF | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF     | CHERIF     | 0000 |
| 12D0                        | DCH3SSA                         | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 12E0                        | DCH3DSA                         | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 12F0                        | DCH3SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1300                        | DCH3DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1310                        | DCH3SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1320                        | DCH3DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1330                        | DCH3CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1340                        | DCH3CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1350                        | DCH3DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1360                        | DCH4CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000       |      |
| 1370                        | DCH4ECON                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 00FF |
|                             |                                 | 15:0      | CHAIRQ<7:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | FF00       |      |
| 1380                        | DCH4INT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIE | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE     | CHERIE     | 0000 |
|                             |                                 | 15:0      | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIF | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF     | CHERIF     | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.



TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits         |       |          |       |          |       |      |      |        |             |        |        |        |        |        | All Resets |        |      |
|-----------------------------|---------------------------------|-----------|--------------|-------|----------|-------|----------|-------|------|------|--------|-------------|--------|--------|--------|--------|--------|------------|--------|------|
|                             |                                 |           | 31/15        | 30/14 | 29/13    | 28/12 | 27/11    | 26/10 | 25/9 | 24/8 | 23/7   | 22/6        | 21/5   | 20/4   | 19/3   | 18/2   | 17/1   |            | 16/0   |      |
| 1390                        | DCH4SSA                         | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13A0                        | DCH4DSA                         | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13B0                        | DCH4SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13C0                        | DCH4DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13D0                        | DCH4SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13E0                        | DCH4DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 13F0                        | DCH4CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1400                        | DCH4CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1410                        | DCH4DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1420                        | DCH5CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | —    | CHCHNS | CHEN        | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000   |      |
| 1430                        | DCH5ECON                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | CHAIRQ<7:0> |        |        |        |        |        | 00FF       |        |      |
|                             |                                 | 15:0      | CHSIRQ<7:0>  |       |          |       |          |       |      |      | CFORCE | CABORT      | PATEN  | SIRQEN | AIRQEN | —      | —      | —          | FF00   |      |
| 1440                        | DCH5INT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | CHSDIE      | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE     | CHERIE | 0000 |
|                             |                                 | 15:0      | —            | —     | —        | —     | —        | —     | —    | —    | —      | CHSDIF      | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF     | CHERIF | 0000 |
| 1450                        | DCH5SSA                         | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1460                        | DCH5DSA                         | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
|                             |                                 | 15:0      |              |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1470                        | DCH5SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1480                        | DCH5DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |
| 1490                        | DCH5SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —    | —      | —           | —      | —      | —      | —      | —      | —          | 0000   |      |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |      |        |             |        |        |        |        |        | 0000       |        |      |

**Legend:** × = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

**TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits         |       |          |       |          |       |      |        |        |        |        |        |        |        |            | All Resets |      |
|-----------------------------|---------------------------------|-----------|--------------|-------|----------|-------|----------|-------|------|--------|--------|--------|--------|--------|--------|--------|------------|------------|------|
|                             |                                 |           | 31/15        | 30/14 | 29/13    | 28/12 | 27/11    | 26/10 | 25/9 | 24/8   | 23/7   | 22/6   | 21/5   | 20/4   | 19/3   | 18/2   | 17/1       |            | 16/0 |
| 14A0                        | DCH5DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 14B0                        | DCH5CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 14C0                        | DCH5CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 14D0                        | DCH5DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 14E0                        | DCH6CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000       |      |
| 14F0                        | DCH6ECON                        | 31:16     | CHAIRQ<7:0>  |       |          |       |          |       |      | —      | —      | —      | —      | —      | —      | —      | —          | —          | 00FF |
|                             |                                 | 15:0      | CHSIRQ<7:0>  |       |          |       |          |       |      | CFORCE | CABORT | PATEN  | SIRQEN | AIRQEN | —      | —      | —          | FF00       |      |
| 1500                        | DCH6INT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIE | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE     | CHERIE     | 0000 |
|                             |                                 | 15:0      | —            | —     | —        | —     | —        | —     | —    | —      | CHSDIF | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF     | CHERIF     | 0000 |
| 1510                        | DCH6SSA                         | 31:16     | CHSSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      | CHSSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1520                        | DCH6DSA                         | 31:16     | CHDSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
|                             |                                 | 15:0      | CHDSA<31:0>  |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1530                        | DCH6SSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1540                        | DCH6DSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1550                        | DCH6SPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1560                        | DCH6DPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1570                        | DCH6CSIZ                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1580                        | DCH6CPTR                        | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 1590                        | DCH6DAT                         | 31:16     | —            | —     | —        | —     | —        | —     | —    | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |          |       |          |       |      |        |        |        |        |        |        |        |            | 0000       |      |
| 15A0                        | DCH7CON                         | 31:16     | CHPIGN<7:0>  |       |          |       |          |       |      | —      | —      | —      | —      | —      | —      | —      | —          | —          | 0000 |
|                             |                                 | 15:0      | CHBUSY       | —     | CHPIGNEN | —     | CHPATLEN | —     | —    | CHCHNS | CHEN   | CHAED  | CHCHN  | CHAEN  | —      | CHEDET | CHPRI<1:0> | 0000       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

**TABLE 9-3: DMA CHANNEL 0 THROUGH CHANNEL 7 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF81_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits         |       |       |       |       |       |      |        |             |        |        |        |        |        |        | All Resets |      |
|-----------------------------|---------------------------------|-----------|--------------|-------|-------|-------|-------|-------|------|--------|-------------|--------|--------|--------|--------|--------|--------|------------|------|
|                             |                                 |           | 31/15        | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8   | 23/7        | 22/6   | 21/5   | 20/4   | 19/3   | 18/2   | 17/1   |            | 16/0 |
| 15B0                        | DCH7ECON                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | CHAIRQ<7:0> |        |        |        |        |        |        | 00FF       |      |
|                             |                                 | 15:0      | CHSIRQ<7:0>  |       |       |       |       |       |      | CFORCE | CABORT      | PATEN  | SIRQEN | AIRQEN | —      | —      | —      | FF00       |      |
| 15C0                        | DCH7INT                         | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | CHSDIE      | CHSHIE | CHDDIE | CHDHIE | CHBCIE | CHCCIE | CHTAIE | CHERIE     | 0000 |
|                             |                                 | 15:0      | —            | —     | —     | —     | —     | —     | —    | —      | CHSDIF      | CHSHIF | CHDDIF | CHDHIF | CHBCIF | CHCCIF | CHTAIF | CHERIF     | 0000 |
| 15D0                        | DCH7SSA                         | 31:16     | CHSSA<31:0>  |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
|                             |                                 | 15:0      | CHSSA<31:0>  |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 15E0                        | DCH7DSA                         | 31:16     | CHDSA<31:0>  |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
|                             |                                 | 15:0      | CHDSA<31:0>  |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 15F0                        | DCH7SSIZ                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHSSIZ<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1600                        | DCH7DSIZ                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHDSIZ<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1610                        | DCH7SPTR                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHSPTR<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1620                        | DCH7DPTR                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHDPTR<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1630                        | DCH7CSIZ                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHCSIZ<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1640                        | DCH7CPTR                        | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHCPTR<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |
| 1650                        | DCH7DAT                         | 31:16     | —            | —     | —     | —     | —     | —     | —    | —      | —           | —      | —      | —      | —      | —      | —      | —          | 0000 |
|                             |                                 | 15:0      | CHPDAT<15:0> |       |       |       |       |       |      |        |             |        |        |        |        |        |        | 0000       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-1: DMACON: DMA CONTROLLER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4         | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0                    | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —                      | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0                    | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —                      | —              | —              | —             | —             |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                  | R/W-0          | U-0            | U-0           | U-0           |
|           | ON             | —              | —              | SUSPEND <sup>(1)</sup> | DMABUSY        | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0                    | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —                      | —              | —              | —             | —             |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** DMA On bit

1 = DMA module is enabled

0 = DMA module is disabled

bit 14-13 **Unimplemented:** Read as '0'

bit 12 **SUSPEND:** DMA Suspend bit<sup>(1)</sup>

1 = DMA transfers are suspended to allow CPU uninterrupted access to data bus

0 = DMA operates normally

bit 11 **DMABUSY:** DMA Module Busy bit

1 = DMA module is active and is transferring data

0 = DMA module is disabled and not actively transferring data

bit 10-0 **Unimplemented:** Read as '0'

**Note 1:** If the user application clears this bit, it may take a number of cycles before the DMA module completes the current transaction and responds to this request. The user application should poll the BUSY bit to verify that the request has been honored.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-2: DMASTAT: DMA STATUS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | RDWR           | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | R-0            | R-0           | R-0           |
|           | —              | —              | —              | —              | —              | DMACH<2:0>     |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **RDWR:** Read/Write Status bit

1 = Last DMA bus access when an error was detected was a read

0 = Last DMA bus access when an error was detected was a write

bit 30-3 **Unimplemented:** Read as '0'

bit 2-0 **DMACH<2:0>:** DMA Channel bits

These bits contain the value of the most recent active DMA channel when an error was detected.

**Note:** The DMASTAT register will be cleared when its contents are read. If more than one errors at the same time, the read transaction will be recorded. Additional transfers that occur later with an error will not update this register until it has been read or cleared.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-3: DMAADDR: DMA ADDRESS REGISTER

| Bit Range      | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| DMAADDR<31:24> |                |                |                |                |                |                |               |               |
| 23:16          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| DMAADDR<23:16> |                |                |                |                |                |                |               |               |
| 15:8           | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| DMAADDR<15:8>  |                |                |                |                |                |                |               |               |
| 7:0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| DMAADDR<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **DMAADDR<31:0>**: DMA Module Address bits

These bits contain the address of the most recent DMA access when an error was detected.

**Note:** The DMAADDR register will be cleared when its contents are read. If more than one errors at the same time, the read transaction will be recorded. Additional transfers that occur later with an error will not update this register until it has been read or cleared.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-4: DCRCCON: DMA CRC CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6        | Bit 29/21/13/5 | Bit 28/20/12/4               | Bit 27/19/11/3     | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|-----------------------|----------------|------------------------------|--------------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0                   | R/W-0          | R/W-0                        | R/W-0              | U-0            | U-0           | R/W-0         |
|           | —              | —                     | BYTO<1:0>      |                              | WBO <sup>(1)</sup> | —              | —             | BITO          |
| 23:16     | U-0            | U-0                   | U-0            | U-0                          | U-0                | U-0            | U-0           | U-0           |
|           | —              | —                     | —              | —                            | —                  | —              | —             | —             |
| 15:8      | U-0            | U-0                   | U-0            | R/W-0                        | R/W-0              | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —                     | —              | PLEN<4:0> <sup>(1,2,3)</sup> |                    |                |               |               |
| 7:0       | R/W-0          | R/W-0                 | R/W-0          | U-0                          | U-0                | R/W-0          | R/W-0         | R/W-0         |
|           | CRCEN          | CRCAPP <sup>(1)</sup> | CRCTYP         | —                            | —                  | CRCCH<2:0>     |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-30 **Unimplemented:** Read as '0'

bit 29-28 **BYTO<1:0>:** CRC Byte Order Selection bits

11 = Endian byte swap on half-word boundaries (i.e., source half-word order with reverse source byte order per half-word)

10 = Swap half-words on word boundaries (i.e., reverse source half-word order with source byte order per half-word)

01 = Endian byte swap on word boundaries (i.e., reverse source byte order)

00 = No swapping (i.e., source byte order)

bit 27 **WBO:** CRC Write Byte Order Selection bit<sup>(1)</sup>

1 = Source data is written to the destination re-ordered as defined by BYTO<1:0>

0 = Source data is written to the destination unaltered

bit 26-25 **Unimplemented:** Read as '0'

bit 24 **BITO:** CRC Bit Order Selection bit

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

1 = The IP header checksum is calculated Least Significant bit (LSb) first (i.e., reflected)

0 = The IP header checksum is calculated Most Significant bit (MSb) first (i.e., not reflected)

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

1 = The LFSR CRC is calculated Least Significant bit first (i.e., reflected)

0 = The LFSR CRC is calculated Most Significant bit first (i.e., not reflected)

bit 23-13 **Unimplemented:** Read as '0'

bit 12-8 **PLEN<4:0>:** Polynomial Length bits<sup>(1,2,3)</sup>

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

These bits are unused.

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

Denotes the length of the polynomial – 1.

bit 7 **CRCEN:** CRC Enable bit

1 = CRC module is enabled and channel transfers are routed through the CRC module

0 = CRC module is disabled and channel transfers proceed normally

**Note 1:** When WBO = 1, unaligned transfers are not supported and the CRCAPP bit cannot be set.

**2:** The maximum CRC length supported by the DMA module is 32.

**3:** This bit is unused when CRCTYP is equal to '1'.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 9-4: DCRCCON: DMA CRC CONTROL REGISTER (CONTINUED)

- bit 6     **CRCAPP**: CRC Append Mode bit<sup>(1)</sup>  
1 = The DMA transfers data from the source into the CRC but NOT to the destination. When a block transfer completes the DMA writes the calculated CRC value to the location given by CHxDSA  
0 = The DMA transfers data from the source through the CRC obeying WBO as it writes the data to the destination
- bit 5     **CRCTYP**: CRC Type Selection bit  
1 = The CRC module will calculate an IP header checksum  
0 = The CRC module will calculate a LFSR CRC
- bit 4-3   **Unimplemented**: Read as '0'
- bit 2-0   **CRCCH<2:0>**: CRC Channel Select bits  
111 = CRC is assigned to Channel 7  
110 = CRC is assigned to Channel 6  
101 = CRC is assigned to Channel 5  
100 = CRC is assigned to Channel 4  
011 = CRC is assigned to Channel 3  
010 = CRC is assigned to Channel 2  
001 = CRC is assigned to Channel 1  
000 = CRC is assigned to Channel 0

- Note 1:** When WBO = 1, unaligned transfers are not supported and the CRCAPP bit cannot be set.  
**2:** The maximum CRC length supported by the DMA module is 32.  
**3:** This bit is unused when CRCTYP is equal to '1'.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-5: DCRCDATA: DMA CRC DATA REGISTER

| Bit Range       | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCDATA<31:24> |                |                |                |                |                |                |               |               |
| 23:16           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCDATA<23:16> |                |                |                |                |                |                |               |               |
| 15:8            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCDATA<15:8>  |                |                |                |                |                |                |               |               |
| 7:0             | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCDATA<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **DCRCDATA<31:0>**: CRC Data Register bits

Writing to this register will seed the CRC generator. Reading from this register will return the current value of the CRC. Bits greater than PLEN will return '0' on any read.

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

Only the lower 16 bits contain IP header checksum information. The upper 16 bits are always '0'. Data written to this register is converted and read back in 1's complement form (i.e., current IP header checksum value).

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

Bits greater than PLEN will return '0' on any read.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-6: DCRCXOR: DMA CRCXOR ENABLE REGISTER

| Bit Range      | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCXOR<31:24> |                |                |                |                |                |                |               |               |
| 23:16          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCXOR<23:16> |                |                |                |                |                |                |               |               |
| 15:8           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCXOR<15:8>  |                |                |                |                |                |                |               |               |
| 7:0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| DCRCXOR<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **DCRCXOR<31:0>**: CRC XOR Register bits

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

This register is unused.

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

1 = Enable the XOR input to the Shift register

0 = Disable the XOR input to the Shift register; data is shifted in directly from the previous stage in the register



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 9-7: DCHxCON: DMA CHANNEL 'x' CONTROL REGISTER ('x' = 0-7) (CONTINUED)

- bit 4     **CHAEN:** Channel Automatic Enable bit  
          1 = Channel is continuously enabled, and not automatically disabled after a block transfer is complete  
          0 = Channel is disabled on block transfer complete
- bit 3     **Unimplemented:** Read as '0'
- bit 2     **CHEDET:** Channel Event Detected bit  
          1 = An event has been detected  
          0 = No events have been detected
- bit 1-0   **CHPRI<1:0>:** Channel Priority bits  
          11 = Channel has priority 3 (highest)  
          10 = Channel has priority 2  
          01 = Channel has priority 1  
          00 = Channel has priority 0

- Note 1:** The chain selection bit takes effect when chaining is enabled (i.e., CHCHN = 1).
- 2:** When the channel is suspended by clearing this bit, the user application should poll the CHBUSY bit (if available on the device variant) to see when the channel is suspended, as it may take some clock cycles to complete a current transaction before the channel is suspended.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 9-8: DCHxECON: DMA CHANNEL x EVENT CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                        | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                          | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | R/W-1                      | R/W-1          | R/W-1          | R/W-1          | R/W-1          | R/W-1          | R/W-1         | R/W-1         |
|           | CHAIRQ<7:0> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 15:8      | R/W-1                      | R/W-1          | R/W-1          | R/W-1          | R/W-1          | R/W-1          | R/W-1         | R/W-1         |
|           | CHSIRQ<7:0> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 7:0       | S-0                        | S-0            | R/W-0          | R/W-0          | R/W-0          | U-0            | U-0           | U-0           |
|           | CFORCE                     | CABORT         | PATEN          | SIRQEN         | AIRQEN         | —              | —             | —             |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | S = Settable bit                   |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31-24 **Unimplemented:** Read as '0'

bit 23-16 **CHAIRQ<7:0>**: Channel Transfer Abort IRQ bits<sup>(1)</sup>

11111111 = Interrupt 255 will abort any transfers in progress and set CHAIF flag

•  
•  
•

00000001 = Interrupt 1 will abort any transfers in progress and set CHAIF flag

00000000 = Interrupt 0 will abort any transfers in progress and set CHAIF flag

bit 15-8 **CHSIRQ<7:0>**: Channel Transfer Start IRQ bits<sup>(1)</sup>

11111111 = Interrupt 255 will initiate a DMA transfer

•  
•  
•

00000001 = Interrupt 1 will initiate a DMA transfer

00000000 = Interrupt 0 will initiate a DMA transfer

**Note:** The DMA does not support I<sup>2</sup>C, Change Notification, Input Capture, CTMU, QEI, and Motor Control PWMs. Use of any of these DMA trigger transfer events could lead to unexpected behavior.

bit 7 **CFORCE:** DMA Forced Transfer bit

1 = A DMA transfer is forced to begin when this bit is written to a '1'

0 = This bit always reads '0'

bit 6 **CABORT:** DMA Abort Transfer bit

1 = A DMA transfer is aborted when this bit is written to a '1'

0 = This bit always reads '0'

bit 5 **PATEN:** Channel Pattern Match Abort Enable bit

1 = Abort transfer and clear CHEN on pattern match

0 = Pattern match is disabled

bit 4 **SIRQEN:** Channel Start IRQ Enable bit

1 = Start channel cell transfer if an interrupt matching CHSIRQ occurs

0 = Interrupt number CHSIRQ is ignored and does not start a transfer

bit 3 **AIRQEN:** Channel Abort IRQ Enable bit

1 = Channel transfer is aborted if an interrupt matching CHAIRQ occurs

0 = Interrupt number CHAIRQ is ignored and does not terminate a transfer

bit 2-0 **Unimplemented:** Read as '0'

**Note 1:** See [Table 6-3: "Interrupt IRQ, Vector and Bit Location"](#) for the list of available interrupt IRQ sources.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-9: DCHxINT: DMA CHANNEL x INTERRUPT CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSDIE         | CHSHIE         | CHDDIE         | CHDHIE         | CHBCIE         | CHCCIE         | CHTAIE        | CHERIE        |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSDIF         | CHSHIF         | CHDDIF         | CHDHIF         | CHBCIF         | CHCCIF         | CHTAIF        | CHERIF        |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-24 **Unimplemented:** Read as '0'

bit 23 **CHSDIE:** Channel Source Done Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 22 **CHSHIE:** Channel Source Half Empty Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 21 **CHDDIE:** Channel Destination Done Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 20 **CHDHIE:** Channel Destination Half Full Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 19 **CHBCIE:** Channel Block Transfer Complete Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 18 **CHCCIE:** Channel Cell Transfer Complete Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 17 **CHTAIE:** Channel Transfer Abort Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 16 **CHERIE:** Channel Address Error Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **CHSDIF:** Channel Source Done Interrupt Flag bit

1 = Channel Source Pointer has reached end of source (CHSPTR = CHSSIZ)

0 = No interrupt is pending

bit 6 **CHSHIF:** Channel Source Half Empty Interrupt Flag bit

1 = Channel Source Pointer has reached midpoint of source (CHSPTR = CHSSIZ/2)

0 = No interrupt is pending

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 9-9: DCHxINT: DMA CHANNEL x INTERRUPT CONTROL REGISTER (CONTINUED)

- bit 5     **CHDDIF**: Channel Destination Done Interrupt Flag bit  
          1 = Channel Destination Pointer has reached end of destination (CHDPTR = CHDSIZ)  
          0 = No interrupt is pending
- bit 4     **CHDHIF**: Channel Destination Half Full Interrupt Flag bit  
          1 = Channel Destination Pointer has reached midpoint of destination (CHDPTR = CHDSIZ/2)  
          0 = No interrupt is pending
- bit 3     **CHBCIF**: Channel Block Transfer Complete Interrupt Flag bit  
          1 = A block transfer has been completed (the larger of CHSSIZ/CHDSIZ bytes has been transferred), or a  
              pattern match event occurs  
          0 = No interrupt is pending
- bit 2     **CHCCIF**: Channel Cell Transfer Complete Interrupt Flag bit  
          1 = A cell transfer has been completed (CHCSIZ bytes have been transferred)  
          0 = No interrupt is pending
- bit 1     **CHTAIF**: Channel Transfer Abort Interrupt Flag bit  
          1 = An interrupt matching CHAIRQ has been detected and the DMA transfer has been aborted  
          0 = No interrupt is pending
- bit 0     **CHERIF**: Channel Address Error Interrupt Flag bit  
          1 = A channel address error has been detected  
              Either the source or the destination address is invalid.  
          0 = No interrupt is pending

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-10: DCHxSSA: DMA CHANNEL x SOURCE START ADDRESS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSA<31:24>   |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSA<23:16>   |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSA<15:8>    |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSA<7:0>     |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0      **CHSSA<31:0>** Channel Source Start Address bits  
Channel source start address.

**Note:** This must be the physical address of the source.

## REGISTER 9-11: DCHxDSEA: DMA CHANNEL x DESTINATION START ADDRESS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSA<31:24>   |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSA<23:16>   |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSA<15:8>    |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSA<7:0>     |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0      **CHDSA<31:0>**: Channel Destination Start Address bits  
Channel destination start address.

**Note:** This must be the physical address of the destination.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 9-12: DCHxSSIZ: DMA CHANNEL x SOURCE SIZE REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSIZ<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHSSIZ<7:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHSSIZ<15:0>**: Channel Source Size bits

1111111111111111 = 65,535 byte source size

.

.

.

0000000000000010 = 2 byte source size

0000000000000001 = 1 byte source size

0000000000000000 = 65,536 byte source size

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-13: DCHxDSIZ: DMA CHANNEL x DESTINATION SIZE REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSIZ<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHDSIZ<7:0>    |                |                |                |                |                |               |               |

### Legend:

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHDSIZ<15:0>**: Channel Destination Size bits

```

111111111111111111 = 65,535 byte destination size
.
.
.
000000000000000010 = 2 byte destination size
000000000000000001 = 1 byte destination size
000000000000000000 = 65,536 byte destination size
    
```



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-15: DCHxDPTR: DMA CHANNEL x DESTINATION POINTER REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | CHDPTR<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | CHDPTR<7:0>    |                |                |                |                |                |               |               |

### Legend:

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHDPTR<15:0>**: Channel Destination Pointer bits

```

1111111111111111 = Points to byte 65,535 of the destination
.
.
.
0000000000000001 = Points to byte 1 of the destination
0000000000000000 = Points to byte 0 of the destination
    
```

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 9-16: DCHxCSIZ: DMA CHANNEL x CELL-SIZE REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHCSIZ<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHCSIZ<7:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHCSIZ<15:0>**: Channel Cell-Size bits

1111111111111111 = 65,535 bytes transferred on an event

·  
·  
·

0000000000000010 = 2 bytes transferred on an event

0000000000000001 = 1 byte transferred on an event

0000000000000000 = 65,536 bytes transferred on an event

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 9-17: DCHxCPTR: DMA CHANNEL x CELL POINTER REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | CHCPTR<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | CHCPTR<7:0>    |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHCPTR<15:0>**: Channel Cell Progress Pointer bits

1111111111111111 = 65,535 bytes have been transferred since the last event  
 .  
 .  
 .  
 0000000000000001 = 1 byte has been transferred since the last event  
 0000000000000000 = 0 bytes have been transferred since the last event

**Note:** When in Pattern Detect mode, this register is reset on a pattern detect.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 9-18: DCHxDAT: DMA CHANNEL x PATTERN DATA REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHPDAT<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CHPDAT<7:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHPDAT<15:0>:** Channel Data Register bits

Pattern Terminate mode:

Data to be matched must be stored in this register to allow terminate on match.

All other modes:

Unused.

# PIC32MK GPG/MCJ with CAN FD Family

## 10.0 I/O PORTS

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 12. “I/O Ports”** (DS60001120), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

General purpose I/O pins are the simplest of peripherals. They allow the PIC32MK GPG/MCJ with CAN FD Family of devices to monitor and control other devices. To add flexibility and functionality, some pins

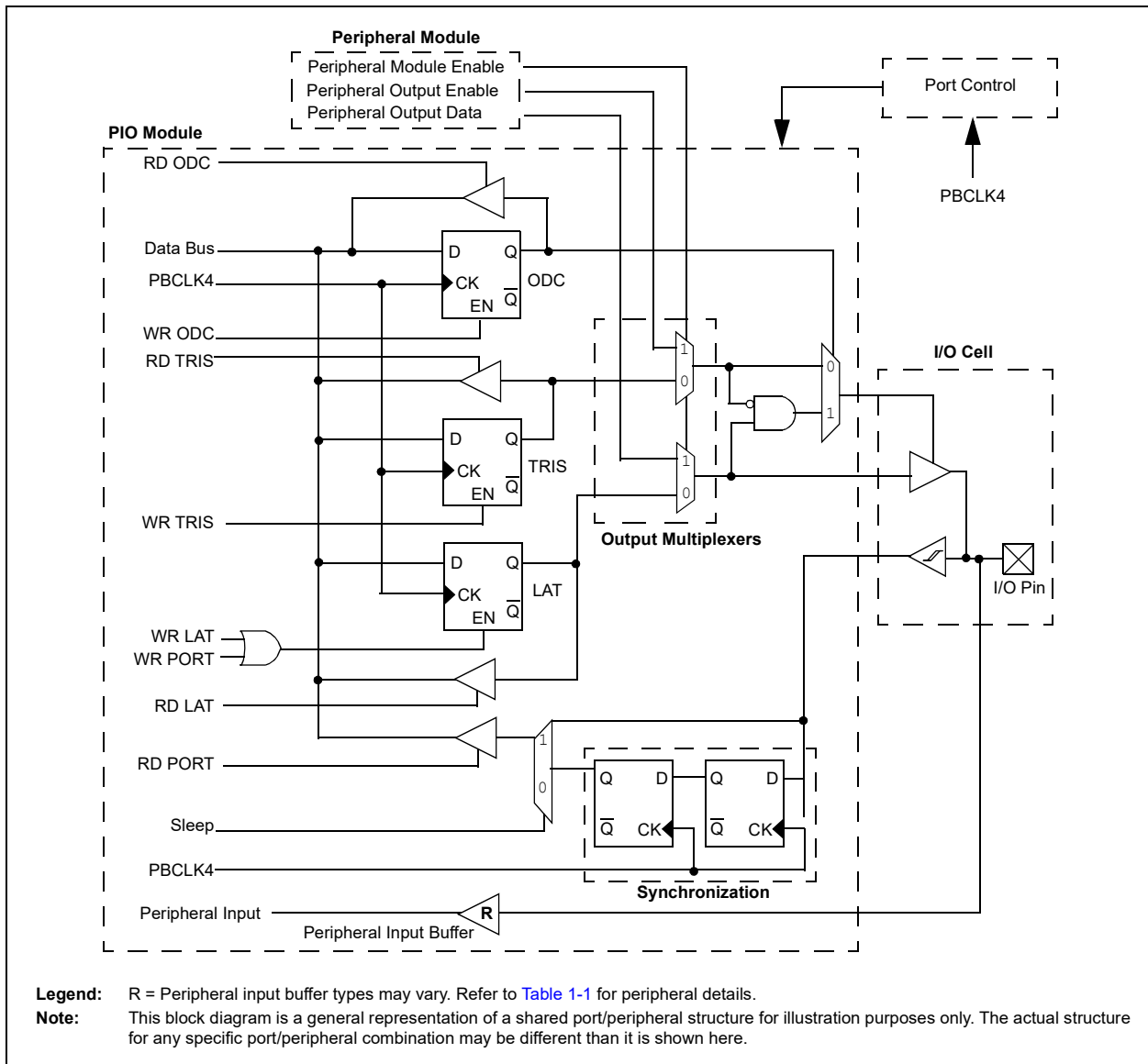
are multiplexed with alternate function(s). These functions depend on which peripheral features are on the device. In general, when a peripheral is functioning, that pin may not be used as a general purpose I/O pin.

The following are key features of the I/O ports:

- Individual output pin open-drain enable/disable
- Individual input pin weak pull-up and pull-down
- Monitor selective inputs and generate interrupt when change in pin state is detected
- Operation during Sleep and Idle modes
- Fast bit manipulation using CLR, SET, and INV registers

Figure 10-1 illustrates a block diagram of a typical multiplexed I/O port.

**FIGURE 10-1: BLOCK DIAGRAM OF A TYPICAL MULTIPLEXED PORT STRUCTURE**





# PIC32MK GPG/MCJ with CAN FD Family

## 10.1 Parallel I/O (PIO) Ports

All port pins have ten registers directly associated with their operation as digital I/O. The data direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the latch (LATx) read the latch. Writes to the latch write the latch. Reads from the port (PORTx) read the port pins, while writes to the port pins write the latch.

### 10.1.1 OPEN-DRAIN CONFIGURATION

In addition to the PORTx, LATx, and TRISx registers for data control, some port pins can also be individually configured for either digital or open-drain output. This is controlled by the Open-Drain Control register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs higher than VDD (e.g., 5V) on any desired 5V-tolerant pins by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum VIH specification.

Refer to the pin name tables ([Table 2](#) and [Table 4](#)) for the available pins and their functionality.

### 10.1.2 CONFIGURING ANALOG AND DIGITAL PORT PINS

The ANSELx register controls the operation of the analog port pins. The port pins that are to function as analog inputs must have their corresponding ANSEL and TRIS bits set. In order to use port pins for I/O functionality with digital modules, such as Timers, UARTs, etc., the corresponding ANSELx bit must be cleared.

The ANSELx register has a default value of 0xFFFF; therefore, all pins that share analog functions are analog (not digital) by default.

If the TRIS bit is cleared (output) while the ANSELx bit is set, the digital output level (VOH or VOL) is converted by an analog peripheral, such as the ADC module or Comparator module.

When the PORT register is read, all pins configured as analog input channels are read as cleared (a low level).

Pins configured as digital inputs do not convert an analog input. Analog levels on any pin defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

### 10.1.3 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically this instruction would be an NOP.

### 10.1.4 INPUT CHANGE NOTIFICATION

The input change notification function of the I/O ports allows the PIC32MK GPG/MCJ with CAN FD Family devices to generate interrupt requests to the processor in response to a change-of-state on selected input pins. This feature can detect input change-of-states even in Sleep mode, when the clocks are disabled. Every I/O port pin can be selected (enabled) for generating an interrupt request on a change-of-state.

Five control registers are associated with the CN functionality of each I/O port. The CNENx and CNNEEx registers contain the CN interrupt enable control bits for each of the input pins. Setting these bits enables a CN interrupt for the corresponding pins. The CNENx register enables a mismatch CN interrupt condition when the EDGEDETECT bit (CNCONx<11>) is not set. When the EDGEDETECT bit is set, the CNNEEx register controls the negative edge while the CNENx register controls the positive edge.

The CNSTATx and CNFx registers indicate the status of change notice based on the setting of the EDGEDETECT bit. If the EDGEDETECT bit is set to '0', the CNSTATx register indicates whether a change occurred on the corresponding pin since the last read of the PORTx bit. If the EDGEDETECT bit is set to '1', the CNFx register indicates whether a change has occurred and through the CNNEEx and CNENx registers the edge type of the change that occurred is also indicated.

Each I/O pin also has a weak pull-up and a weak pull-down connected to it. The pull-ups act as a current source or sink source connected to the pin, and eliminate the need for external resistors when push-button or keypad devices are connected. The pull-ups and pull-downs are enabled separately using the CNPUx and the CNPDx registers, which contain the control bits for each of the pins. Setting any of the control bits enables the weak pull-ups and/or pull-downs for the corresponding pins.

**Note:** Pull-ups and pull-downs on change notification pins should always be disabled when the port pin is configured as a digital output.

An additional control register (CNCONx) is shown in [Register 10-3](#).

# PIC32MK GPG/MCJ with CAN FD Family

---

## 10.2 CLR, SET, and INV Registers

Every I/O module register has a corresponding CLR (clear), SET (set) and INV (invert) register designed to provide fast atomic bit manipulations. As the name of the register implies, a value written to a SET, CLR or INV register effectively performs the implied operation, but only on the corresponding base register and only bits specified as '1' are modified. Bits specified as '0' are not modified.

Reading SET, CLR and INV registers returns undefined values. To see the affects of a write operation to a SET, CLR or INV register, the base register must be read.

## 10.3 Peripheral Pin Select (PPS)

A major challenge in general purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. The challenge is even greater on low pin-count devices. In an application where more than one peripheral needs to be assigned to a single pin, inconvenient workarounds in application code or a complete redesign may be the only option.

PPS configuration provides an alternative to these choices by enabling peripheral set selection and their placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, users can better tailor the device to their entire application, rather than trimming the application to fit the device.

The PPS configuration feature operates over a fixed subset of digital I/O pins. Users may independently map the input and/or output of most digital peripherals to these I/O pins. PPS is performed in software and generally does not require the device to be reprogrammed. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping once it has been established.

### 10.3.1 AVAILABLE PINS

The number of available pins is dependent on the particular device and its pin count. Pins that support the PPS feature include the designation "RPn" in their full pin designation, where "RP" designates a remappable peripheral and "n" is the remappable port number.

### 10.3.2 AVAILABLE PERIPHERALS

The peripherals managed by the PPS are all digital-only peripherals. These include general serial communications (UART, SPI, and CAN), general purpose timer clock inputs, timer-related peripherals (input capture and output compare), interrupt-on-change inputs, and reference clocks (input and output).

In comparison, some digital-only peripheral modules are never included in the PPS feature. This is because the peripheral's function requires special I/O circuitry on a specific port and cannot be easily connected to multiple pins. A similar requirement excludes all modules with analog inputs, such as the Analog-to-Digital Converter (ADC).

A key difference between remappable and non-remappable peripherals is that remappable peripherals are not associated with a default I/O pin. The peripheral must always be assigned to a specific I/O pin before it can be used. In contrast, non-remappable peripherals are always available on a default pin, assuming that the peripheral is active and not conflicting with another peripheral.

When a remappable peripheral is active on a given I/O pin, it takes priority over all other digital I/O and digital communication peripherals associated with the pin. Priority is given regardless of the type of peripheral that is mapped. Remappable peripherals never take priority over any analog functions associated with the pin.

### 10.3.3 CONTROLLING PPS

PPS features are controlled through two sets of SFRs: one to map peripheral inputs, and one to map outputs. Because they are separately controlled, a particular peripheral's input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral-selectable pin is handled in two different ways, depending on whether an input or output is being mapped.

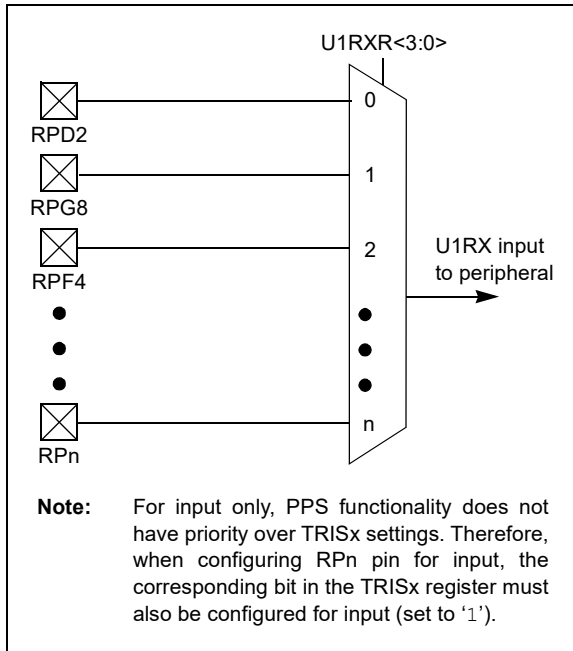
### 10.3.4 INPUT MAPPING

The inputs of the PPS options are mapped on the basis of the peripheral. That is, a control register associated with a peripheral dictates the pin it will be mapped to. The *[pin name]R* registers, where *[pin name]* refers to the peripheral pins listed in [Table 10-1](#), are used to configure peripheral input mapping (see [Register 10-1](#)). Each register contains sets of 4 bit fields. Programming these bit fields with an appropriate value maps the RPn pin with the corresponding value to that peripheral. For any given device, the valid range of values for any bit field is shown in [Table 10-1](#).

[Figure 10-2](#) illustrates the remappable pin selection for the U1RX input.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 10-2: REMAPPABLE INPUT  
EXAMPLE FOR U1RX**



# PIC32MK GPG/MCJ with CAN FD Family

TABLE 10-1: INPUT PIN SELECTION

| Peripheral Pin       | [pin name]R SFR            | [pin name]R bits      | [pin name]R Value to RPN Pin Selection |
|----------------------|----------------------------|-----------------------|--|
| INT4                 | INT4R<3:0>                 | INT4R                 | 0000 = RPA0                            |
| T2CK                 | T2CKR<3:0>                 | T2CKR                 | 0001 = RPB3                            |
| T6CK                 | T6CKR<3:0>                 | T6CKR                 | 0010 = RPB4                            |
| IC4                  | IC4R<3:0>                  | IC4R                  | 0011 = RPB15                           |
| IC7                  | IC7R<3:0>                  | IC7R                  | 0100 = RPB7                            |
| SDI1                 | SDI1R<3:0>                 | SDI1R                 | 0101 = RPC7 <sup>(2)</sup>             |
| QEA1 <sup>(3)</sup>  | QEA1R<3:0> <sup>(3)</sup>  | QEA1R <sup>(3)</sup>  | 0110 = RPC0                            |
| HOME2 <sup>(3)</sup> | HOME2R<3:0> <sup>(3)</sup> | HOME2R <sup>(3)</sup> | 0111 = Reserved                        |
| QAEA3 <sup>(3)</sup> | QAEA3R<3:0> <sup>(3)</sup> | QEA3R <sup>(3)</sup>  | 1000 = RPA11 <sup>(2)</sup>            |
| FLT1 <sup>(3)</sup>  | FLT1R<3:0> <sup>(3)</sup>  | FLT1R <sup>(3)</sup>  | 1001 = RPD5 <sup>(1)</sup>             |
| REFCLKI              | REFIR <3:0>                | REFIR                 | 1010 = RPG6 <sup>(1)</sup>             |
| FLT2 <sup>(3)</sup>  | FLT2R<3:0> <sup>(3)</sup>  | FLT2R <sup>(3)</sup>  | 1011 = RPF1 <sup>(1)</sup>             |
| CLCINC               | CLCINCR<3:0>               | CLCINCR               | 1100 = RPE0 <sup>(1)</sup>             |
|                      |                            |                       | 1101 = RPA15 <sup>(1)</sup>            |
|                      |                            |                       | 1110 = Reserved                        |
|                      |                            |                       | 1111 = Reserved                        |

- Note 1:** 64-Pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on “MCJ” variant.  
**Note 4:** Not available on “GPG” variant.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-1: INPUT PIN SELECTION (CONTINUED)**

| Peripheral Pin       | [pin name]R SFR            | [pin name]R bits      | [pin name]R Value to RPn Pin Selection |
|----------------------|----------------------------|-----------------------|--|
| INT3                 | INT3R<3:0>                 | INT3R                 | 0000 = RPA1                            |
| T3CK                 | T3CKR<3:0>                 | T3CKR                 | 0001 = RPB5                            |
| T7CK                 | T7CKR<3:0>                 | T7CKR                 | 0010 = RPB1                            |
| IC3                  | IC3R<3:0>                  | IC3R                  | 0011 = RPB11                           |
| IC8                  | IC8R<3:0>                  | IC8R                  | 0100 = RPB8                            |
| U1CTS                | U1CTSR<3:0>                | U1CTSR                | 0101 = RPA8                            |
| U2RX                 | U2RXR<3:0>                 | U2RXR                 | 0110 = RPC8 <sup>(2)</sup>             |
| SDI2                 | SDI2R<3:0>                 | SDI2R                 | 0111 = RPB12                           |
| QEB1 <sup>(3)</sup>  | QEB1R<3:0> <sup>(3)</sup>  | QEB1R <sup>(3)</sup>  | 1000 = RPA12 <sup>(2)</sup>            |
| INDX2 <sup>(3)</sup> | INDX2R<3:0> <sup>(3)</sup> | INDX2R <sup>(3)</sup> | 1001 = RPD6 <sup>(1)</sup>             |
| QEB3 <sup>(3)</sup>  | QEB3R<3:0> <sup>(3)</sup>  | QEB3R <sup>(3)</sup>  | 1010 = RPG7 <sup>(1)</sup>             |
| FLT2 <sup>(3)</sup>  | FLT2R <3:0> <sup>(3)</sup> | FLT2R <sup>(3)</sup>  | 1011 = Reserved                        |
| CLCINA               | CLCINAR<3:0>               | CLCINAR               | 1100 = RPE1 <sup>(1)</sup>             |
|                      |                            |                       | 1101 = RPA14 <sup>(1)</sup>            |
|                      |                            |                       | 1110 = Reserved                        |
|                      |                            |                       | 1111 = Reserved                        |

- Note 1:** 64-Pin devices only.  
**2:** Register is write only.  
**3:** Only available on “MCJ” variant.  
**4:** Not available on “GPG” variant.

# PIC32MK GPG/MCJ with CAN FD Family

TABLE 10-1: INPUT PIN SELECTION (CONTINUED)

| Peripheral Pin       | [pin name]R SFR            | [pin name]R bits      | [pin name]R Value to RPN Pin Selection |
|----------------------|----------------------------|-----------------------|--|
| INT2                 | INT2R<3:0>                 | INT2R                 | 0000 = RPB6                            |
| T4CK                 | T4CKR<3:0>                 | T4CKR                 | 0001 = RPC15                           |
| T8CK                 | T8CKR<3:0>                 | T8CKR                 | 0010 = RPA4                            |
| IC1                  | IC1R<3:0>                  | IC1R                  | 0011 = RPB13                           |
| IC5                  | IC5R<3:0>                  | IC5R                  | 0100 = RPB2                            |
| IC9                  | IC9R<3:0>                  | IC9R                  | 0101 = RPC6 <sup>(2)</sup>             |
| U1RX                 | U1RXR<3:0>                 | U1RXR                 | 0110 = RPC1                            |
| U2CTS                | U2CTSR<3:0>                | U2CTSR                | 0111 = RPA7                            |
| SS1                  | SS1R<3:0>                  | SS1R                  | 1000 = RPE14 <sup>(1)</sup>            |
| SCK2                 | SCK2R[3:0]                 | SCK2R                 | 1001 = RPC13                           |
| INDX1 <sup>(3)</sup> | INDX1R<3:0> <sup>(3)</sup> | INDX1R <sup>(3)</sup> | 1010 = RPG8 <sup>(1)</sup>             |
| QEB2 <sup>(3)</sup>  | QEB2R<3:0> <sup>(3)</sup>  | QEB2R <sup>(3)</sup>  | 1011 = Reserved                        |
| INDX3 <sup>(3)</sup> | INDX3R<3:0> <sup>(3)</sup> | INDX3R <sup>(3)</sup> | 1100 = RPF0 <sup>(1)</sup>             |
| C1RX <sup>(4)</sup>  | C1RXR<3:0> <sup>(4)</sup>  | C1RXR <sup>(4)</sup>  | 1101 = Reserved                        |
| OCFB                 | OCFBR<3:0>                 | OCFBR                 | 1110 = Reserved                        |
| CLCIND               | CLCINDR<3:0>               | CLCINDR               | 1111 = Reserved                        |

- Note 1:** 64-Pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on “MCJ” variant.  
**Note 4:** Not available on “GPG” variant.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-1: INPUT PIN SELECTION (CONTINUED)**

| Peripheral Pin       | [pin name]R SFR            | [pin name]R bits      | [pin name]R Value to RPN Pin Selection |
|----------------------|----------------------------|-----------------------|--|
| INT1                 | INT1R<3:0>                 | INT1R                 | 0000 = RPB14                           |
| T5CK                 | T5CKR<3:0>                 | T5CKR                 | 0001 = RPC12                           |
| T9CK                 | T9CKR<3:0>                 | T9CKR                 | 0010 = RPB0                            |
| IC2                  | IC2R<3:0>                  | IC2R                  | 0011 = RPB10                           |
| IC6                  | IC6R<3:0>                  | IC6R                  | 0100 = RPB9                            |
| SS2                  | SS2R<3:0> <sup>(2)</sup>   | SS2R                  | 0101 = RPC9 <sup>(2)</sup>             |
| HOME1 <sup>(3)</sup> | HOME1R<3:0> <sup>(3)</sup> | HOME1R <sup>(3)</sup> | 0110 = RPC2 <sup>(2)</sup>             |
| QEA2 <sup>(3)</sup>  | QEA2R<3:0> <sup>(3)</sup>  | QEA2R <sup>(3)</sup>  | 0111 = Reserved                        |
| HOME3 <sup>(3)</sup> | HOME3R<3:0> <sup>(3)</sup> | HOME3R <sup>(3)</sup> | 1000 = RPE15 <sup>(1)</sup>            |
| OCFA                 | OCFAR<3:0>                 | OCFAR                 | 1001 = RPC10                           |
| CLCINB               | CLCINBR<3:0>               | CLCINBR               | 1010 = RPG9 <sup>(1)</sup>             |
|                      |                            |                       | 1011 = Reserved                        |
|                      |                            |                       | 1100 = Reserved                        |
|                      |                            |                       | 1101 = Reserved                        |
|                      |                            |                       | 1110 = Reserved                        |
|                      |                            |                       | 1111 = Reserved                        |

- Note 1:** 64-Pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on “MCJ” variant.  
**Note 4:** Not available on “GPG” variant.

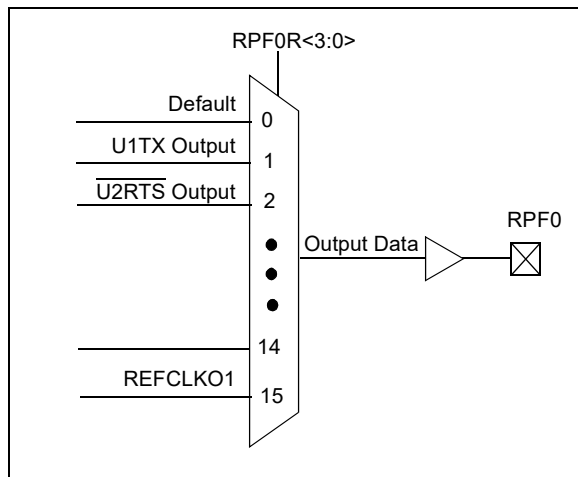
# PIC32MK GPG/MCJ with CAN FD Family

## 10.3.5 OUTPUT MAPPING

In contrast to inputs, the outputs of the PPS options are mapped on the basis of the pin. In this case, a control register associated with a particular pin dictates the peripheral output to be mapped. The RPNR registers (Register 10-2) are used to control output mapping. Like the [pin name]R registers, each register contains sets of 4 bit fields. The value of the bit field corresponds to one of the peripherals, and that peripheral's output is mapped to the pin (see Table 10-2 and Figure 10-3).

A null output is associated with the output register reset value of '0'. This is done to ensure that remappable outputs remain disconnected from all output pins by default.

**FIGURE 10-3: EXAMPLE OF MULTIPLEXING OF REMAPPABLE OUTPUT FOR RPF0**



## 10.3.6 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. The PIC32MK GPG/MCJ with CAN FD Family devices include two features to prevent alterations to the peripheral map:

- Control register lock sequence
- Configuration bit select lock

### 10.3.6.1 Control Register Lock

Under normal operation, writes to the RPNR and [pin name]R registers are not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK Configuration bit (CFGCON<13>). Setting the IOLOCK bit prevents writes to the control registers and clearing the IOLOCK bit allows writes.

To set or clear the IOLOCK bit, an unlock sequence must be executed. Refer to the **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the "PIC32 Family Reference Manual" for details.

### 10.3.6.2 Configuration Bit Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPNR and [pin name]R registers. The IOL1WAY Configuration bit (DEVCFG3<29>) blocks the IOLOCK bit from being cleared after it has been set once. If the IOLOCK bit remains set, the register unlock procedure does not execute, and the PPS control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session.



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-2: OUTPUT PIN SELECTION**

| RPn Port Pin         | RPnR SFR | RPnR bits   | RPnR Value to Peripheral Selection |
|----------------------|----------|-------------|------------------------------------|
| RPA0                 | RPA0R    | RPA0R<4:0>  | 00000 = Off                        |
| RPB3                 | RPB3R    | RPB3R<4:0>  | 00001 = U1TX                       |
| RPB4                 | RPB4R    | RPB4R<4:0>  | 00010 = U2RTS                      |
| RPB15                | RPB15R   | RPB15R<4:0> | 00011 = SDO1                       |
| RPB7                 | RPB7R    | RPB7R<4:0>  | 00100 = SDO2                       |
| RPC7                 | RPC7R    | RPC7R<4:0>  | 00101 = OCI                        |
| RPC0                 | RPC0R    | RPC0R<4:0>  | 00110 = OC7                        |
| RPA11                | RPA11R   | RPA11R<4:0> | 00111 = C2OUT                      |
| RPD5 <sup>(1)</sup>  | RPD5R    | RPD5R<4:0>  | 01000 = C4OUT                      |
| RPG6 <sup>(1)</sup>  | RPG6R    | RPG6R<4:0>  | 01001 = Reserved                   |
| RPF1 <sup>(1)</sup>  | RPF1R    | RPF1R<4:0>  | 01010 = Reserved                   |
| RPE0 <sup>(1)</sup>  | RPE0R    | RPE0R<4:0>  | 01011 = Reserved                   |
| RPA15 <sup>(1)</sup> | RPA15R   | RPA15R<4:0> | 01100 = C1TX <sup>(4)</sup>        |
|                      |          |             | 01101 = Reserved                   |
|                      |          |             | 01110 = Reserved                   |
|                      |          |             | ...                                |
|                      |          |             | 10001 = Reserved                   |
|                      |          |             | 10010 = REFCLKO4                   |
|                      |          |             | 10011 = Reserved                   |
|                      |          |             | 10100 = QEICMP1 <sup>(3)</sup>     |
|                      |          |             | 10101 = Reserved                   |
|                      |          |             | 10110 = Reserved                   |
|                      |          |             | 10111 = Reserved                   |
|                      |          |             | 11000 = CLCO2                      |
|                      |          |             | ...                                |
|                      |          |             | 11111 = Reserved                   |

- Note 1:** 64 pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on “MCJ” variant.  
**Note 4:** Not available on “GPG” variant.  
**Note 5:** SPI Frame Sync Output in Frame Master mode

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-2: OUTPUT PIN SELECTION (CONTINUED)**

| RPn Port Pin         | RPnR SFR | RPnR bits   | RPnR Value to Peripheral Selection |
|----------------------|----------|-------------|------------------------------------|
| RPA1                 | RPA1R    | RPA1R<4:0>  | 00000 = Off                        |
| RPB5                 | RPB5R    | RPB5R<4:0>  | 00001 = Reserved                   |
| RPB1                 | RPB1R    | RPB1R<4:0>  | 00010 = Reserved                   |
| RPB11                | RPB11R   | RPB11R<4:0> | 00011 = SDO1                       |
| RPA8                 | RPA8R    | RPA8R<4:0>  | 00100 = SDO2                       |
| RPC8                 | RPC8R    | RPC8R<4:0>  | 00101 = OC2                        |
| RPB12                | RPB12R   | RPB12R<4:0> | 00110 = OC8                        |
| RPA12                | RPA12R   | RPA12R<4:0> | 00111 = C3OUT                      |
| RPD6 <sup>(1)</sup>  | RPD6R    | RPD6R<4:0>  | 01000 = OC9                        |
| RPG7 <sup>(1)</sup>  | RPG7R    | RPG7R<4:0>  | 01001 = Reserved                   |
| RPE1 <sup>(1)</sup>  | RPE1R    | RPE1R<4:0>  | ***                                |
| RPA14 <sup>(1)</sup> | RPA14R   | RPA14R<4:0> | 10001 = Reserved                   |
|                      |          |             | 10010 = REFOUT3                    |
|                      |          |             | 10011 = Reserved                   |
|                      |          |             | 10100 = QEICMP2 <sup>(3)</sup>     |
|                      |          |             | 10101 = Reserved                   |
|                      |          |             | 10110 = Reserved                   |
|                      |          |             | 10111 = Reserved                   |
|                      |          |             | 11000 = CLCO1                      |
|                      |          |             | 11001 = Reserved                   |
|                      |          |             | ***                                |
|                      |          |             | 11111 = Reserved                   |

- Note 1:** 64 pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on "MCJ" variant.  
**Note 4:** Not available on "GPG" variant.  
**Note 5:** SPI Frame Sync Output in Frame Master mode

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-2: OUTPUT PIN SELECTION (CONTINUED)**

| RPn Port Pin         | RPnR SFR | RPnR bits   | RPnR Value to Peripheral Selection   |
|----------------------|----------|-------------|--|
| RPB6                 | RPB6R    | RPB6R<4:0>  | 00000 = Off<br>00001 = Reserved  |
| RPC15                | RPC15R   | RPC15R<4:0> | 00010 = Reserved<br>00011 = SS1 <sup>(5)</sup>   |
| RPA4                 | RPA4R    | RPA4R<4:0>  | 00100 = SCK2<br>00101 = OC4  |
| RPB13                | RPB13R   | RPB13R<4:0> | 00110 = OC5<br>00111 = REFOUT1   |
| RPB2                 | RPB2R    | RPB2R<4:0>  | 01000 = C5OUT<br>01001 = Reserved  |
| RPC6                 | RPC6R    | RPC6R<4:0>  | ***  |
| RPC1                 | RPC1R    | RPC1R<4:0>  | 10001 = Reserved<br>10010 = REFOUT2<br>10011 = Reserved                                    |
| RPA7                 | RPA7R    | RPA7R<4:0>  | 10100 = QEICMP3 <sup>(3)</sup><br>10101 = Reserved<br>10110 = Reserved<br>10111 = Reserved |
| RPE14 <sup>(1)</sup> | RPE14R   | RPE14R<4:0> | 11000 = CLCO4<br>11001 = Reserved  |
| RPG8 <sup>(1)</sup>  | RPG8R    | RPG8R<4:0>  | ***  |
| RPF0 <sup>(1)</sup>  | RPF0R    | RPF0R<4:0>  | 11111 = Reserved   |

- Note 1:** 64 pin devices only.  
**2:** Register is write only.  
**3:** Only available on “MCJ” variant.  
**4:** Not available on “GPG” variant.  
**5:** SPI Frame Sync Output in Frame Master mode

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 10-2: OUTPUT PIN SELECTION (CONTINUED)**

| RPn Port Pin         | RPnR SFR | RPnR bits   | RPnR Value to Peripheral Selection             |
|----------------------|----------|-------------|--|
| RPB14                | RPB14R   | RPB14R<4:0> | 00000 = Off                                    |
| RPC12                | RPC12R   | RPC12R<4:0> | 00001 = U1RTS<br>00010 = U2TX                  |
| RPB0                 | RPB0R    | RPB0R<4:0>  | 00011 = Reserved<br>00100 = SS2 <sup>(5)</sup> |
| RPB10                | RPB10R   | RPB10R<4:0> | 00101 = OC3<br>00110 = OC6                     |
| RPB9                 | RPB9R    | RPB9R<4:0>  | 00111 = C1OUT<br>01000 = Reserved              |
| RPC9                 | RPC9R    | RPC9R<4:0>  | 01001 = Reserved<br>* * *                      |
| RPC2                 | RPC2R    | RPC2R<4:0>  | 10111 = Reserved                               |
| RPE15 <sup>(1)</sup> | RPE15R   | RPE15R<4:0> | 11000 = CLCO3<br>11001 = Reserved              |
| RPC10                | RPC10R   | RPC10R<4:0> | * * *<br>11110 = Reserved                      |
| RPG9 <sup>(1)</sup>  | RPG9R    | RPG9R<4:0>  | 11111 = Reserved                               |

- Note 1:** 64 pin devices only.  
**Note 2:** Register is write only.  
**Note 3:** Only available on “MCJ” variant.  
**Note 4:** Not available on “GPG” variant.  
**Note 5:** SPI Frame Sync Output in Frame Master mode

## 10.4 I/O Ports Control Registers

**TABLE 10-3: PORTA REGISTER MAP**

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits                          |                               |       |                |                |                |      |               |               |      |      |      |               |      |      |               | All<br>Resets |
|-----------------------------|---------------------------------|-----------|-------------------------------|-------------------------------|-------|----------------|----------------|----------------|------|---------------|---------------|------|------|------|---------------|------|------|---------------|---------------|
|                             |                                 |           | 31/15                         | 30/14                         | 29/13 | 28/12          | 27/11          | 26/10          | 25/9 | 24/8          | 23/7          | 22/6 | 21/5 | 20/4 | 19/3          | 18/2 | 17/1 | 16/0          |               |
| 0000                        | ANSELA                          | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | ANSA47 <sup>(1)</sup>         | ANSA46 <sup>(1)</sup>         | —     | ANSA10         | ANSA9          | —              | —    | ANSA26        | —             | —    | —    | —    | ANSA24        | —    | —    | ANSA1         | ANSA0         |
| 0010                        | TRISA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | TRISA15 <sup>(1)</sup>        | TRISA14 <sup>(1)</sup>        | —     | TRISA12        | TRISA11        | TRISA10        | —    | TRISA8        | TRISA7        | —    | —    | —    | TRISA4        | —    | —    | TRISA1        | TRISA0        |
| 0020                        | PORTA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | RA15 <sup>(1)</sup>           | RA14 <sup>(1)</sup>           | —     | RA12           | RA11           | RA10           | —    | RA8           | RA7           | —    | —    | —    | RA4           | —    | —    | RA1           | RA0           |
| 0030                        | LATA                            | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | LATA15 <sup>(1)</sup>         | LATA14 <sup>(1)</sup>         | —     | LATA12         | LATA11         | LATA10         | —    | LATA8         | LATA7         | —    | —    | —    | LATA4         | —    | —    | LATA1         | LATA0         |
| 0040                        | ODCA                            | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | ODCA15 <sup>(1)</sup>         | ODCA14 <sup>(1)</sup>         | —     | ODCA12         | ODCA11         | ODCA10         | —    | ODCA8         | ODCA7         | —    | —    | —    | ODCA4         | —    | —    | ODCA1         | ODCA0         |
| 0050                        | CNPUA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | CNPUA15 <sup>(1)</sup>        | CNPUA14 <sup>(1)</sup>        | —     | CNPUA12        | CNPUA11        | CNPUA10        | —    | CNPUA8        | CNPUA7        | —    | —    | —    | CNPUA4        | —    | —    | CNPUA1        | CNPUA0        |
| 0060                        | CNPDA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | CNP-<br>DA15 <sup>(1)</sup>   | CNP-<br>DA14 <sup>(1)</sup>   | —     | CNPDA12        | CNPDA11        | CNPDA10        | —    | CNPDA8        | CNPDA7        | —    | —    | —    | CNPDA4        | —    | —    | CNPDA1        | CNPDA0        |
| 0070                        | CNCONA                          | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | ON                            | —                             | SIDL  | —              | EDGE<br>DETECT | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | —             |
| 0080                        | CNENA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | CNE-<br>NA15 <sup>(1)</sup>   | CNE-<br>NA14 <sup>(1)</sup>   | —     | CNENA12        | CNENA11        | CNENA10        | —    | CNENA8        | CNENA7        | —    | —    | —    | CNENA4        | —    | —    | CNENA1        | CNENA0        |
| 0090                        | CNSTATA                         | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | CNSTA-<br>TA15 <sup>(1)</sup> | CNSTA-<br>TA14 <sup>(1)</sup> | —     | CNSTA-<br>TA12 | CNSTA-<br>TA11 | CNSTA-<br>TA10 | —    | CNSTA-<br>TA8 | CNSTA-<br>TA7 | —    | —    | —    | CNSTA-<br>TA4 | —    | —    | CNSTA-<br>TA1 | CNSTA-<br>TA0 |
| 00A0                        | CNNEA                           | 31:16     | —                             | —                             | —     | —              | —              | —              | —    | —             | —             | —    | —    | —    | —             | —    | —    | —             | 0000          |
|                             |                                 | 15:0      | CNNEA15 <sup>(1)</sup>        | CNNEA14 <sup>(1)</sup>        | —     | CNNEA12        | CN<br>NEA11    | CNNEA10        | —    | CNNEA8        | CNNEA7        | —    | —    | —    | CNNEA4        | —    | —    | CNNEA1        | CNNEA0        |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48-pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits          |               |       |        |        |        |      |       |       |      |      |       |      |      |       |       | All<br>Resets |
|-----------------------------|---------------------------------|-----------|---------------|---------------|-------|--------|--------|--------|------|-------|-------|------|------|-------|------|------|-------|-------|---------------|
|                             |                                 |           | 31/15         | 30/14         | 29/13 | 28/12  | 27/11  | 26/10  | 25/9 | 24/8  | 23/7  | 22/6 | 21/5 | 20/4  | 19/3 | 18/2 | 17/1  | 16/0  |               |
| 00B0                        | CNFA                            | 31:16     | —             | —             | —     | —      | —      | —      | —    | —     | —     | —    | —    | —     | —    | —    | —     | —     | 0000          |
|                             |                                 | 15:0      | CNFA15<br>(1) | CNFA14<br>(1) | —     | CNFA12 | CNFA11 | CNFA10 | —    | CNFA8 | CNFA7 | —    | —    | CNFA4 | —    | —    | CNFA1 | CNFA0 | 0000          |
| 00C0                        | SRCON0A                         | 31:16     | —             | —             | —     | —      | —      | —      | —    | —     | —     | —    | —    | —     | —    | —    | —     | —     | 0000          |
|                             |                                 | 15:0      | —             | —             | —     | —      | —      | SR0A10 | —    | SR0A8 | SR0A7 | —    | —    | —     | —    | —    | SR0A1 | —     | 0000          |
| 00D0                        | SRCON1A                         | 31:16     | —             | —             | —     | —      | —      | —      | —    | —     | —     | —    | —    | —     | —    | —    | —     | —     | 0000          |
|                             |                                 | 15:0      | —             | —             | —     | —      | —      | SR1A10 | —    | SR1A8 | SR1A7 | —    | —    | —     | —    | —    | SR1A1 | —     | 0000          |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48-pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-4: PORTB REGISTER MAP

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits          |               |               |               |                |               |              |        |              |              |              |              |              |              |              | All<br>Resets |       |
|-----------------------------|---------------------------------|-----------|---------------|---------------|---------------|---------------|----------------|---------------|--------------|--------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|---------------|-------|
|                             |                                 |           | 31/15         | 30/14         | 29/13         | 28/12         | 27/11          | 26/10         | 25/9         | 24/8   | 23/7         | 22/6         | 21/5         | 20/4         | 19/3         | 18/2         | 17/1         |               | 16/0  |
| 0100                        | ANSELB                          | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | —             | —             | —             | —             | —              | —             | —            | ANSA27 | —            | ANSA25       | —            | —            | —            | ANSA5        | ANSA4        | ANSA3         | ANSA2 |
| 0110                        | TRISB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | TRISB15       | TRISB14       | TRISB13       | TRISB12       | TRISB11        | TRISB10       | TRISB9       | —      | TRISB7       | TRISB6       | TRISB5       | TRISB4       | TRISB3       | TRISB2       | TRISB1       | TRISB0        | FFFF  |
| 0120                        | PORTB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | RB15          | RB14          | RB13          | RB12          | RB11           | RB10          | RB9          | RB8    | RB7          | RB6          | RB5          | RB4          | RB3          | RB2          | RB1          | RB0           | xxxx  |
| 0130                        | LATB                            | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | LATB15        | LATB14        | LATB13        | LATB12        | LATB11         | LATB10        | LATB9        | —      | LATB7        | LATB6        | LATB5        | LATB4        | LATB3        | LATB2        | LATB1        | LATB0         | xxxx  |
| 0140                        | ODCB                            | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | ODCB15        | ODCB14        | ODCB13        | ODCB12        | ODCB11         | ODCB10        | ODCB9        | —      | ODCB7        | ODCB6        | ODCB5        | ODCB4        | ODCB3        | ODCB2        | ODCB1        | ODCB0         | 0000  |
| 0150                        | CNPUB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNPUB15       | CNPUB14       | CNPUB13       | CNPUB12       | CNPUB11        | CNPUB10       | CNPUB9       | —      | CNPUB7       | CNPUB6       | CNPUB5       | CNPUB4       | CNPUB3       | CNPUB2       | CNPUB1       | CNPUB0        | 0000  |
| 0160                        | CNPDB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNPDB15       | CNPDB14       | CNPDB13       | CNPDB12       | CNPDB11        | CNPDB10       | CNPDB9       | —      | CNPDB7       | CNPDB6       | CNPDB5       | CNPDB4       | CNPDB3       | CNPDB2       | CNPDB1       | CNPDB0        | 0000  |
| 0170                        | CNCONB                          | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | ON            | —             | SIDL          | —             | EDGE<br>DETECT | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
| 0180                        | CNENB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNENB15       | CNENB14       | CNENB13       | CNENB12       | CNENB11        | CNENB10       | CNENB9       | —      | CNENB7       | CNENB6       | CNENB5       | CNENB4       | CNENB3       | CNENB2       | CNENB1       | CNENB0        | 0000  |
| 0190                        | CNSTATB                         | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNSTATB<br>15 | CNSTATB<br>14 | CNSTATB<br>13 | CNSTATB<br>12 | CNSTATB<br>11  | CNSTATB<br>10 | CNSTATB<br>9 | —      | CNSTATB<br>7 | CNSTATB<br>6 | CNSTATB<br>5 | CNSTATB<br>4 | CNSTATB<br>3 | CNSTATB<br>2 | CNSTATB<br>1 | CNSTATB<br>0  | 0000  |
| 01A0                        | CNNEB                           | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNNEB15       | CNNEB14       | CNNEB13       | CNNEB12       | CNNEB11        | CNNEB10       | CNNEB9       | —      | CNNEB7       | CNNEB6       | CNNEB5       | CNNEB4       | CNNEB3       | CNNEB2       | CNNEB1       | CNNEB0        | 0000  |
| 01B0                        | CNFB                            | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | CNFB15        | CNFB14        | CNFB13        | CNFB12        | CNFB11         | CNFB10        | CNFB9        | —      | CNFB7        | CNFB6        | CNFB5        | CNFB4        | CNFB3        | CNFB2        | CNFB1        | CNFB0         | 0000  |
| 01C0                        | SRCON0B                         | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | SR0B15        | SR0B14        | SR0B13        | SR0B12        | SR0B11         | SR0B10        | —            | —      | SR0B7        | SR0B6        | SR0B5        | SR0B4        | —            | —            | —            | —             | 0000  |
| 01D0                        | SRCON1B                         | 31:16     | —             | —             | —             | —             | —              | —             | —            | —      | —            | —            | —            | —            | —            | —            | —            | —             | 0000  |
|                             |                                 | 15:0      | SR1B15        | SR1B14        | SR1B13        | SR1B12        | SR1B11         | SR1B10        | —            | —      | SR1B7        | SR1B6        | SR1B5        | SR1B4        | —            | —            | —            | —             | 0000  |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-5: PORTC REGISTER MAP

| Virtual Address<br>(BF86_#) | Register<br>Name(1) | Bit Range | Bits      |       |       |           |                |           |          |          |          |          |      |      |      |      |          | All<br>Resets |          |
|-----------------------------|---------------------|-----------|-----------|-------|-------|-----------|----------------|-----------|----------|----------|----------|----------|------|------|------|------|----------|---------------|----------|
|                             |                     |           | 31/15     | 30/14 | 29/13 | 28/12     | 27/11          | 26/10     | 25/9     | 24/8     | 23/7     | 22/6     | 21/5 | 20/4 | 19/3 | 18/2 | 17/1     |               | 16/0     |
| 0200                        | ANSEL               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | 0000          |          |
|                             |                     | 15:0      | —         | —     | —     | ANSA49    | ANSA11         | ANSA48    | —        | —        | —        | —        | —    | —    | —    | —    | ANSA8    | ANSA7         | ANSA6    |
| 0210                        | TRISC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | TRISC15   | —     | —     | TRISC12   | TRISC11        | TRISC10   | TRISC9   | TRISC8   | TRISC7   | TRISC6   | —    | —    | —    | —    | TRISC2   | TRISC1        | TRISC0   |
| 0220                        | PORTC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | RC15      | —     | RC13  | RC12      | RC11           | RC10      | RC9      | RC8      | RC7      | RC6      | —    | —    | —    | —    | RC2      | RC1           | RC0      |
| 0230                        | LATC                | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | LATC15    | —     | —     | LATC12    | LATC11         | LATC10    | LATC9    | LATC8    | LATC7    | LATC6    | —    | —    | —    | —    | LATC2    | LATC1         | LATC0    |
| 0240                        | ODCC                | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | ODCC15    | —     | —     | ODCC12    | ODCC11         | ODCC10    | ODCC9    | ODCC8    | ODCC7    | ODCC6    | —    | —    | —    | —    | ODCC2    | ODCC1         | ODCC0    |
| 0250                        | CNPUC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNPUC15   | —     | —     | CNPUC12   | CNPUC11        | CNPUC10   | CNPUC9   | CNPUC8   | CNPUC7   | CNPUC6   | —    | —    | —    | —    | CNPUC2   | CNPUC1        | CNPUC0   |
| 0260                        | CNPDC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNPDC15   | —     | —     | CNPDC12   | CNPDC11        | CNPDC10   | CNPDC9   | CNPDC8   | CNPDC7   | CNPDC6   | —    | —    | —    | —    | CNPDC2   | CNPDC1        | CNPDC0   |
| 0270                        | CNCONC              | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | ON        | —     | SIDL  | —         | EDGE<br>DETECT | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | —        |
| 0280                        | CNENC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNENC15   | —     | —     | CNENC12   | CNENC11        | CNENC10   | CNENC9   | CNENC8   | CNENC7   | CNENC6   | —    | —    | —    | —    | CNENC2   | CNENC1        | CNENC0   |
| 0290                        | CNSTATC             | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNSTATC15 | —     | —     | CNSTATC12 | CNSTATC11      | CNSTATC10 | CNSTATC9 | CNSTATC8 | CNSTATC7 | CNSTATC6 | —    | —    | —    | —    | CNSTATC2 | CNSTATC1      | CNSTATC0 |
| 02A0                        | CNNEC               | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNNEC15   | —     | —     | CNNEC12   | CNNEC11        | CNNEC10   | CNNEC9   | CNNEC8   | CNNEC7   | CNNEC6   | —    | —    | —    | —    | CNNEC2   | CNNEC1        | CNNEC0   |
| 02B0                        | CNFC                | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | CNFC15    | —     | —     | CNFC12    | CNFC11         | CNFC10    | CNFC9    | CNFC8    | CNFC7    | CNFC6    | —    | —    | —    | —    | CNFC2    | CN—FC1        | CNFC0    |
| 02C0                        | SRCON0<br>C         | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | SR0C15    | —     | —     | —         | SR0C11         | —         | SR0C9    | SR0C8    | SR0C7    | SR0C6    | —    | —    | —    | —    | —        | —             | —        |
| 02D0                        | SRCON1<br>C         | 31:16     | —         | —     | —     | —         | —              | —         | —        | —        | —        | —        | —    | —    | —    | —    | —        | —             | 0000     |
|                             |                     | 15:0      | SR1C15    | —     | —     | —         | SR1C11         | —         | SR1C9    | SR1C8    | SR1C7    | SR1C6    | —    | —    | —    | —    | —        | —             | —        |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only Available on "MC" variants.



**TABLE 10-6: PORTD REGISTER MAP**

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |       |                |       |              |      |                             |                             |      |      |      |      |      | All<br>Resets |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|-------|----------------|-------|--------------|------|-----------------------------|-----------------------------|------|------|------|------|------|---------------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11          | 26/10 | 25/9         | 24/8 | 23/7                        | 22/6                        | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |               | 16/0 |
| 0300                        | ANSELD                          | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
| 0310                        | TRISD                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0160 |
| 0320                        | PORTD                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | RD8          | —    | RD6 <sup>(1)</sup>          | RD5 <sup>(1)</sup>          | —    | —    | —    | —    | —    | —             | xxxx |
| 0330                        | LATD                            | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | LATD8        | —    | LATD6 <sup>(1)</sup>        | LATD5 <sup>(1)</sup>        | —    | —    | —    | —    | —    | —             | xxxx |
| 0340                        | ODCD                            | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | ODCD8        | —    | ODCD6 <sup>(1)</sup>        | ODCD5 <sup>(1)</sup>        | —    | —    | —    | —    | —    | —             | 0000 |
| 0350                        | CNPUD                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNPUD8       | —    | CNPUD6 <sup>(1)</sup>       | CNPUD5 <sup>(1)</sup>       | —    | —    | —    | —    | —    | —             | 0000 |
| 0360                        | CNPDD                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNPDD8       | —    | CNPDD6 <sup>(1)</sup>       | CNPDD5 <sup>(1)</sup>       | —    | —    | —    | —    | —    | —             | 0000 |
| 0370                        | CNCOND                          | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | ON    | —     | SIDL  | —     | EDGE<br>DETECT | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
| 0380                        | CNEND                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNEND8       | —    | CNEND6 <sup>(1)</sup>       | CNEND5 <sup>(1)</sup>       | —    | —    | —    | —    | —    | —             | 0000 |
| 0390                        | CNSTATD                         | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNSTAT<br>D8 | —    | CNSTAT<br>D6 <sup>(1)</sup> | CNSTAT<br>D5 <sup>(1)</sup> | —    | —    | —    | —    | —    | —             | 0000 |
| 03A0                        | CNNED                           | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNNED8       | —    | CNNED6 <sup>(1)</sup>       | CNNED5 <sup>(1)</sup>       | —    | —    | —    | —    | —    | —             | 0000 |
| 03B0                        | CNFD                            | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | CNFD8        | —    | CNFD6 <sup>(1)</sup>        | CNFD5 <sup>(1)</sup>        | —    | —    | —    | —    | —    | —             | 0000 |
| 03C0                        | SRCON0D                         | 31:16     | —     | —     | —     | —     | —              | —     | —            | —    | —                           | —                           | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | SR0D8        | —    | SR0D6 <sup>(1)</sup>        | SR0D5 <sup>(1)</sup>        | —    | —    | —    | —    | —    | —             | 0000 |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note** 1: Does not exist on 48 pin variants.  
 2: Not available on "GPG" variants.  
 3: Only available on "MC" variants.

| Virtual Address<br>(BF86_#) | Register<br>Name(1) | Bit Range | Bits  |       |       |       |       |       |      |       |      |                      |                      |      |      |      |      |      | All<br>Resets |
|-----------------------------|---------------------|-----------|-------|-------|-------|-------|-------|-------|------|-------|------|----------------------|----------------------|------|------|------|------|------|---------------|
|                             |                     |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8  | 23/7 | 22/6                 | 21/5                 | 20/4 | 19/3 | 18/2 | 17/1 | 16/0 |               |
| 03D0                        | SRCON1D             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —     | —    | —                    | —                    | —    | —    | —    | —    | —    | 0000          |
|                             |                     | 15:0      | —     | —     | —     | —     | —     | —     | —    | SR1D8 | —    | SR1D6 <sup>(1)</sup> | SR1D5 <sup>(1)</sup> | —    | —    | —    | —    | —    | —             |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-7: PORTE REGISTER MAP

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit<br>Range | Bits                         |                              |                              |                              |                |       |      |      |      |      |      |      |      |      |                             | All<br>Resets               |      |
|-----------------------------|---------------------------------|--------------|------------------------------|------------------------------|------------------------------|------------------------------|----------------|-------|------|------|------|------|------|------|------|------|-----------------------------|-----------------------------|------|
|                             |                                 |              | 31/15                        | 30/14                        | 29/13                        | 28/12                        | 27/11          | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1                        |                             | 16/0 |
| 0400                        | ANSELE                          | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | ANSA15 <sup>(1)</sup>        | ANSA14 <sup>(1)</sup>        | ANSA13 <sup>(1)</sup>        | ANSA12 <sup>(1)</sup>        | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | ANSA41 <sup>(1)</sup>       | ANSA40 <sup>(1)</sup>       | F003 |
| 0410                        | TRISE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | TRISE15 <sup>(1)</sup>       | TRISE14 <sup>(1)</sup>       | TRISE13 <sup>(1)</sup>       | TRISE12 <sup>(1)</sup>       | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | TRISE1 <sup>(1)</sup>       | TRISE0 <sup>(1)</sup>       | F003 |
| 0420                        | PORTE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | RE15 <sup>(1)</sup>          | RE14 <sup>(1)</sup>          | RE13 <sup>(1)</sup>          | RE12 <sup>(1)</sup>          | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | RE1 <sup>(1)</sup>          | RE0 <sup>(1)</sup>          | xxxx |
| 0430                        | LATE                            | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | LATE15 <sup>(1)</sup>        | LATE14 <sup>(1)</sup>        | LATE13 <sup>(1)</sup>        | LATE12 <sup>(1)</sup>        | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | LATE1 <sup>(1)</sup>        | LATE0 <sup>(1)</sup>        | xxxx |
| 0440                        | ODCE                            | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | ODCE15 <sup>(1)</sup>        | ODCE14 <sup>(1)</sup>        | ODCE13 <sup>(1)</sup>        | ODCE12 <sup>(1)</sup>        | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | ODCE1 <sup>(1)</sup>        | ODCE0 <sup>(1)</sup>        | 0000 |
| 0450                        | CNPUE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNPUE15 <sup>(1)</sup>       | CNPUE14 <sup>(1)</sup>       | CNPUE13 <sup>(1)</sup>       | CNPUE12 <sup>(1)</sup>       | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNPUE1 <sup>(1)</sup>       | CNPUE0 <sup>(1)</sup>       | 0000 |
| 0460                        | CNPDE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNP-<br>DE15 <sup>(1)</sup>  | CNP-<br>DE14 <sup>(1)</sup>  | CNP-<br>DE13 <sup>(1)</sup>  | CNP-<br>DE12 <sup>(1)</sup>  | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNP-<br>DE1 <sup>(1)</sup>  | CNP-<br>DE0 <sup>(1)</sup>  | 0000 |
| 0470                        | CNCONE                          | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | ON                           | —                            | SIDL                         | —                            | EDGE<br>DETECT | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
| 0480                        | CNENE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNENE15 <sup>(1)</sup>       | CNENE14 <sup>(1)</sup>       | CNENE13 <sup>(1)</sup>       | CNENE12 <sup>(1)</sup>       | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNENE1 <sup>(1)</sup>       | CNENE0 <sup>(1)</sup>       | 0000 |
| 0490                        | CNSTATE                         | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNSTATE<br>15 <sup>(1)</sup> | CNSTATE<br>14 <sup>(1)</sup> | CNSTATE<br>13 <sup>(1)</sup> | CNSTATE<br>12 <sup>(1)</sup> | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNSTATE<br>1 <sup>(1)</sup> | CNSTA<br>TE0 <sup>(1)</sup> | 0000 |
| 04A0                        | CNNEE                           | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNNEE15 <sup>(1)</sup>       | CNNEE14 <sup>(1)</sup>       | CNNEE13 <sup>(1)</sup>       | CNNEE12 <sup>(1)</sup>       | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNNEE1 <sup>(1)</sup>       | CNNEE0 <sup>(1)</sup>       | 0000 |
| 04B0                        | CNFE                            | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | CNFE15 <sup>(1)</sup>        | CNFE14 <sup>(1)</sup>        | CNFE13 <sup>(1)</sup>        | CNFE12 <sup>(1)</sup>        | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNFE1 <sup>(1)</sup>        | CNFE0 <sup>(1)</sup>        | 0000 |
| 04C0                        | SRCON0E                         | 31:16        | —                            | —                            | —                            | —                            | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |
|                             |                                 | 15:0         | SR0E15 <sup>(1)</sup>        | SR0E14 <sup>(1)</sup>        | SR0E13 <sup>(1)</sup>        | SR0E12 <sup>(1)</sup>        | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                           | —                           | 0000 |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note** 1: Does not exist on 48 pin variants.  
 2: Not available on "GPG" variants.  
 3: Only available on "MC" variants.

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits                  |                       |                       |                       |       |       |      |      |      |      |      |      |      |      |      | All<br>Resets |      |
|-----------------------------|---------------------------------|-----------|-----------------------|-----------------------|-----------------------|-----------------------|-------|-------|------|------|------|------|------|------|------|------|------|---------------|------|
|                             |                                 |           | 31/15                 | 30/14                 | 29/13                 | 28/12                 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |               | 16/0 |
| 04D0                        | SRCON1E                         | 31:16     | —                     | —                     | —                     | —                     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                                 | 15:0      | SR1E15 <sup>(1)</sup> | SR1E14 <sup>(1)</sup> | SR1E13 <sup>(1)</sup> | SR1E12 <sup>(1)</sup> | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | —    |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-8: PORTF REGISTER MAP

| Virtual Address<br>(BF86_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |       |                |       |      |      |      |      |      |      |      |      |                         | All<br>Resets           |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|-------|----------------|-------|------|------|------|------|------|------|------|------|-------------------------|-------------------------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11          | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1                    |                         | 16/0 |
| 0500                        | ANSELF                          | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
| 0510                        | TRISF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | TRISF1 <sup>(1)</sup>   | TRISF0 <sup>(1)</sup>   | 0003 |
| 0520                        | PORTF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | RF1 <sup>(1)</sup>      | RF0 <sup>(1)</sup>      | xxxx |
| 0530                        | LATF                            | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | LATF1 <sup>(1)</sup>    | LATF0 <sup>(1)</sup>    | xxxx |
| 0540                        | ODCF                            | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | ODCF1 <sup>(1)</sup>    | ODCF0 <sup>(1)</sup>    | 0000 |
| 0550                        | CNPUF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNPUF1 <sup>(1)</sup>   | CNPUF0 <sup>(1)</sup>   | 0000 |
| 0560                        | CNPDF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNPDF1 <sup>(1)</sup>   | CNPDF0 <sup>(1)</sup>   | 0000 |
| 0570                        | CNCONF                          | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | ON    | —     | SIDL  | —     | EDGE<br>DETECT | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
| 0580                        | CNENF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNENF1 <sup>(1)</sup>   | CNENF0 <sup>(1)</sup>   | 0000 |
| 0590                        | CNSTATF                         | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNSTATF1 <sup>(1)</sup> | CNSTATF0 <sup>(1)</sup> | 0000 |
| 05A0                        | CNEEF                           | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNEEF1 <sup>(1)</sup>   | CNEEF0 <sup>(1)</sup>   | 0000 |
| 05B0                        | CNFF                            | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | CNFF1 <sup>(1)</sup>    | CNFF0 <sup>(1)</sup>    | 0000 |
| 05C0                        | SRCON0F                         | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | SR0F1 <sup>(1)</sup>    | SR0F0 <sup>(1)</sup>    | 0000 |
| 05D0                        | SRCON1F                         | 31:16     | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | —                       | —                       | 0000 |
|                             |                                 | 15:0      | —     | —     | —     | —     | —              | —     | —    | —    | —    | —    | —    | —    | —    | —    | SR1F1 <sup>(1)</sup>    | SR1F0 <sup>(1)</sup>    | 0000 |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note** 1: Does not exist on 48 pin variants.  
 2: Not available on "GPG" variants.  
 3: Only available on "MC" variants.

**TABLE 10-9: PORTG REGISTER MAP**

| Virtual Address (BF86_#) | Register Name <sup>(1)</sup> | Bit Range | Bits  |       |       |       |             |       |      |                         |                         |                         |                         |      |      |      |      |      | All Resets |
|--------------------------|------------------------------|-----------|-------|-------|-------|-------|-------------|-------|------|-------------------------|-------------------------|-------------------------|-------------------------|------|------|------|------|------|------------|
|                          |                              |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11       | 26/10 | 25/9 | 24/8                    | 23/7                    | 22/6                    | 21/5                    | 20/4 | 19/3 | 18/2 | 17/1 | 16/0 |            |
| 0600                     | ANSELG                       | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | ANSA16 <sup>(1)</sup>   | ANSA17 <sup>(1)</sup>   | ANSA18 <sup>(1)</sup>   | ANSA19 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 03C0       |
| 0610                     | TRISG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | TRISG9 <sup>(1)</sup>   | TRISG8 <sup>(1)</sup>   | TRISG7 <sup>(1)</sup>   | TRISG6 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 03C0       |
| 0620                     | PORTG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | RG9 <sup>(1)</sup>      | RG8 <sup>(1)</sup>      | RG7 <sup>(1)</sup>      | RG6 <sup>(1)</sup>      | —    | —    | —    | —    | —    | xxxx       |
| 0630                     | LATG                         | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | LATG9 <sup>(1)</sup>    | LATG8 <sup>(1)</sup>    | LATG7 <sup>(1)</sup>    | LATG6 <sup>(1)</sup>    | —    | —    | —    | —    | —    | xxxx       |
| 0640                     | ODCG                         | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | ODCG9 <sup>(1)</sup>    | ODCG8 <sup>(1)</sup>    | ODCG7 <sup>(1)</sup>    | ODCG6 <sup>(1)</sup>    | —    | —    | —    | —    | —    | 0000       |
| 0650                     | CNPUG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNPUG9 <sup>(1)</sup>   | CNPUG8 <sup>(1)</sup>   | CNPUG7 <sup>(1)</sup>   | CNPUG6 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 0000       |
| 0660                     | CNPDG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNPDG9 <sup>(1)</sup>   | CNPDG8 <sup>(1)</sup>   | CNPDG7 <sup>(1)</sup>   | CNPDG6 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 0000       |
| 0670                     | CNCONG                       | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | ON    | —     | SIDL  | —     | EDGE DETECT | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
| 0680                     | CNENG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNENG9 <sup>(1)</sup>   | CNENG8 <sup>(1)</sup>   | CNENG7 <sup>(1)</sup>   | CNENG6 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 0000       |
| 0690                     | CNSTATG                      | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNSTATG9 <sup>(1)</sup> | CNSTATG8 <sup>(1)</sup> | CNSTATG7 <sup>(1)</sup> | CNSTATG6 <sup>(1)</sup> | —    | —    | —    | —    | —    | 0000       |
| 06A0                     | CNNEG                        | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNNEG9 <sup>(1)</sup>   | CNNEG8 <sup>(1)</sup>   | CNNEG7 <sup>(1)</sup>   | CNNEG6 <sup>(1)</sup>   | —    | —    | —    | —    | —    | 0000       |
| 06B0                     | CNFG                         | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | CNFG9 <sup>(1)</sup>    | CNFG8 <sup>(1)</sup>    | CNFG7 <sup>(1)</sup>    | CNFG6 <sup>(1)</sup>    | —    | —    | —    | —    | —    | 0000       |
| 06C0                     | SRCON0G                      | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | —                       | SR0G7 <sup>(1)</sup>    | SR0G6 <sup>(1)</sup>    | —                       | —    | —    | —    | —    | —    | 0000       |
| 06D0                     | SRCON1G                      | 31:16     | —     | —     | —     | —     | —           | —     | —    | —                       | —                       | —                       | —                       | —    | —    | —    | —    | —    | 0000       |
|                          |                              | 15:0      | —     | —     | —     | —     | —           | —     | —    | —                       | SR1G7 <sup>(1)</sup>    | SR1G6 <sup>(1)</sup>    | —                       | —    | —    | —    | —    | —    | 0000       |

**Legend:** x = Unknown value on Reset; — = Unimplemented, read as '0'; Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

**TABLE 10-10: PERIPHERAL PIN SELECT INPUT REGISTER MAP**

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |            |      | All Resets |      |
|-----------------------------|------------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------------|------|------------|------|
|                             |                  |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2       | 17/1 |            | 16/0 |
| 1404                        | INT1R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | INT1R<3:0> |      |            | 0000 |
| 1408                        | INT2R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | INT2R<3:0> |      |            | 0000 |
| 140C                        | INT3R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | INT3R<3:0> |      |            | 0000 |
| 1410                        | INT4R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | INT4R<3:0> |      |            | 0000 |
| 1418                        | T2CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T2CKR<3:0> |      |            | 0000 |
| 141C                        | T3CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T3CKR<3:0> |      |            | 0000 |
| 1420                        | T4CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T4CKR<3:0> |      |            | 0000 |
| 1424                        | T5CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T5CKR<3:0> |      |            | 0000 |
| 1428                        | T6CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T6CKR<3:0> |      |            | 0000 |
| 142C                        | T7CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T7CKR<3:0> |      |            | 0000 |
| 1430                        | T8CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T8CKR<3:0> |      |            | 0000 |
| 1434                        | T9CKR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | T9CKR<3:0> |      |            | 0000 |
| 1438                        | IC1R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | IC1R<3:0>  |      |            | 0000 |
| 143C                        | IC2R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | IC2R<3:0>  |      |            | 0000 |
| 1440                        | IC3R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | IC3R<3:0>  |      |            | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-10: PERIPHERAL PIN SELECT INPUT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | All Resets  |
|-----------------------------|------------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-------------|
|                             |                  |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |             |
| 1444                        | IC4R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC4R<3:0>   |
| 1448                        | IC5R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC5R<3:0>   |
| 144C                        | IC6R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC6R<3:0>   |
| 1450                        | IC7R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC7R<3:0>   |
| 1454                        | IC8R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC8R<3:0>   |
| 1458                        | IC9R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | IC9R<3:0>   |
| 145C                        | OCFAR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | OCFAR<3:0>  |
| 1460                        | OCFBR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | OCFBR<3:0>  |
| 1464                        | U1RXR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | U1RXR<3:0>  |
| 1468                        | U1CTSR           | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | U1CTSR<3:0> |
| 146C                        | U2RXR            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | U2RXR<3:0>  |
| 1470                        | U2CTSR           | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | U2CTSR<3:0> |
| 1498                        | SDI1R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | SDI1R<3:0>  |
| 149C                        | SS1R             | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | SS1R<3:0>   |
| 14A0                        | SCK2R            | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | 0000        |
|                             |                  | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | SCK2R<3:0>  |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: Does not exist on 48 pin variants.  
 2: Not available on "GPG" variants.  
 3: Only available on "MC" variants.



**TABLE 10-10: PERIPHERAL PIN SELECT INPUT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF80_#) | Register Name         | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |             |      | All Resets |      |
|-----------------------------|-----------------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|-------------|------|------------|------|
|                             |                       |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3        | 18/2 |            | 17/1 |
| 14A4                        | SDI2R                 | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | SDI2R<3:0>  |      |            | 0000 |
| 14A8                        | SS2R                  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | SS2R<3:0>   |      |            | 0000 |
| 14C4                        | C1RXR <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | C1RXR<3:0>  |      |            | 0000 |
| 14C8                        | C2RXR <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | C2RXR<3:0>  |      |            | 0000 |
| 14CC                        | REFIR                 | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | REFIR<3:0>  |      |            | 0000 |
| 14D0                        | QEA1R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEA1R<3:0>  |      |            | 0000 |
| 14D4                        | QEB1R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEB1R<3:0>  |      |            | 0000 |
| 14D8                        | INDX1R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | INDX1R<3:0> |      |            | 0000 |
| 14DC                        | HOME1R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | HOME1R<3:0> |      |            | 0000 |
| 14E0                        | QEA2R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEA2R<3:0>  |      |            | 0000 |
| 14E4                        | QEB2R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEB2R<3:0>  |      |            | 0000 |
| 14E8                        | INDX2R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | INDX2R<3:0> |      |            | 0000 |
| 14EC                        | HOME2R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | HOME2R<3:0> |      |            | 0000 |
| 14F0                        | FLT1R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | FLT1R<3:0>  |      |            | 0000 |
| 14F4                        | FLT2R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —           | —    | —          | 0000 |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | FLT2R<3:0>  |      |            | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-10: PERIPHERAL PIN SELECT INPUT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF80_#) | Register Name         | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |              |      |      | All Resets |
|-----------------------------|-----------------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|--------------|------|------|------------|
|                             |                       |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3         | 18/2 | 17/1 |            |
| 152C                        | C3RXR <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | C3RXR<3:0>   |      |      | 0000       |
| 1530                        | C4RXR <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | C4RXR<3:0>   |      |      | 0000       |
| 1534                        | QEA3R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEA3R<3:0>   |      |      | 0000       |
| 1538                        | QEB3R <sup>(3)</sup>  | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | QEB3R<3:0>   |      |      | 0000       |
| 153C                        | INDX3R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | INDX3R<3:0>  |      |      | 0000       |
| 1540                        | HOME3R <sup>(3)</sup> | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | HOME3R<3:0>  |      |      | 0000       |
| 1574                        | CLCIN1                | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | CLCIN1R<3:0> |      |      | 0000       |
| 1578                        | CLCIN2                | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | CLCIN2R<3:0> |      |      | 0000       |
| 157C                        | CLCIN3                | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | CLCIN3R<3:0> |      |      | 0000       |
| 1580                        | CLCIN4                | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —            | —    | —    | 0000       |
|                             |                       | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | CLCIN4R<3:0> |      |      | 0000       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: Does not exist on 48 pin variants.
  - 2: Not available on "GPG" variants.
  - 3: Only available on "MC" variants.

TABLE 10-11: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit<br>Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | All<br>Resets |             |
|-----------------------------|------------------|--------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|---------------|-------------|
|                             |                  |              | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |               | 16/0        |
| 1600                        | RPA0R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA0R<4:0>  |
| 1604                        | RPA1R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA1R<4:0>  |
| 1608                        | RPA2R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA2R<4:0>  |
| 160C                        | RPA3R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA3R<4:0>  |
| 1610                        | RPA4R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA4R<4:0>  |
| 161C                        | RPA7R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA7R<4:0>  |
| 1620                        | RPA8R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA8R<4:0>  |
| 162C                        | RPA11R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA11R<4:0> |
| 1630                        | RPA12R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA12R<4:0> |
| 1638                        | RPA14R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA14R<4:0> |
| 163C                        | RPA15R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPA15R<4:0> |
| 1640                        | RPB0R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB0R<4:0>  |
| 1644                        | RPB1R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB1R<4:0>  |
| 1648                        | RPB2R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB2R<4:0>  |
| 164C                        | RPB3R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB3R<4:0>  |
| 1650                        | RPB4R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB4R<4:0>  |
| 1654                        | RPB5R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB5R<4:0>  |
| 1658                        | RPB6R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000        |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | RPB6R<4:0>  |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** Does not exist on 48 pin variants.

**TABLE 10-11: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF80_#) | Register Name | Bit Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |      |             |
|-----------------------------|---------------|-----------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|------|-------------|
|                             |               |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            | 17/1 | 16/0        |
| 165C                        | RPB7R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB7R<4:0>  |
| 1664                        | RPB9R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB9R<4:0>  |
| 1668                        | RPB10R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB10R<4:0> |
| 166C                        | RPB11R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB11R<4:0> |
| 1670                        | RPB12R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB12R<4:0> |
| 1674                        | RPB13R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB13R<4:0> |
| 1678                        | RPB14R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB14R<4:0> |
| 167C                        | RPB15R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPB15R<4:0> |
| 1680                        | RPC0R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC0R<4:0>  |
| 1684                        | RPC1R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC1R<4:0>  |
| 1688                        | RPC2R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC2R<4:0>  |
| 1690                        | RPC4R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC4R<4:0>  |
| 1698                        | RPC6R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC6R<4:0>  |
| 169C                        | RPC7R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC7R<4:0>  |
| 16A0                        | RPC8R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC8R<4:0>  |
| 16A4                        | RPC9R         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC9R<4:0>  |
| 16A8                        | RPC10R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC10R<4:0> |
| 16B0                        | RPC12R        | 31:16     | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | 0000        |
|                             |               | 15:0      | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —          | —    | RPC12R<4:0> |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** Does not exist on 48 pin variants.

TABLE 10-11: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit<br>Range | Bits  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | All<br>Resets |      |
|-----------------------------|------------------|--------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|---------------|------|
|                             |                  |              | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |               | 16/0 |
| 16BC                        | RPC15R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 16CC                        | RPD3R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 16D0                        | RPD4R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 16D4                        | RPD5R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 16D8                        | RPD6R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1700                        | RPE0R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1704                        | RPE1R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1738                        | RPE14R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 173C                        | RPE15R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1740                        | RPF0R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1744                        | RPF1R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1780                        | RPG0R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1784                        | RPG1R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 1798                        | RPG6R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 179C                        | RPG7R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 17A0                        | RPG8R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 17A4                        | RPG9R            | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
| 17B0                        | RPG12R           | 31:16        | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |
|                             |                  | 15:0         | —     | —     | —     | —     | —     | —     | —    | —    | —    | —    | —    | —    | —    | —    | —    | —             | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** Does not exist on 48 pin variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 10-1: *[pin name]*R: PERIPHERAL PIN SELECT INPUT REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3           | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|--------------------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0                      | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —                        | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0                      | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —                        | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0                      | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —                        | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | <i>[pin name]</i> R<3:0> |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-4 **Unimplemented:** Read as '0'

bit 3-0 ***[pin name]*R<3:0>**: Peripheral Pin Select Input bits

Where *[pin name]* refers to the pins that are used to configure peripheral input mapping. See [Table 10-1](#) for input pin selection values.

**Note:** Register values can only be changed if the IOLOCK Configuration bit (CFGCON<13>) = 0.

## REGISTER 10-2: RPnR: PERIPHERAL PIN SELECT OUTPUT REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | RPnR<4:0>      |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-5 **Unimplemented:** Read as '0'

bit 4-0 **RPnR<4:0>**: Peripheral Pin Select Output bits

See [Table 10-2](#) for output pin selection values.

**Note:** Register values can only be changed if the IOLOCK Configuration bit (CFGCON<13>) = 0.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 10-3: CNCONx: CHANGE NOTICE CONTROL FOR PORTx REGISTER (x = A – G)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | U-0            | R/W-0          | U-0            | R/W-0          | r-0            | U-0           | U-0           |
|           | ON             | —              | SIDL           | —              | EDGEDETECT     | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

- bit 31-16 **Unimplemented:** Read as '0'
- bit 15 **ON:** Change Notice (CN) Control ON bit
  - 1 = CN is enabled
  - 0 = CN is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **SIDL:** Stop in Idle Control bit
  - 1 = CPU Idle mode halts CN operation
  - 0 = CPU Idle mode does not affect CN operation
- bit 12 **Unimplemented:** Read as '0'
- bit 11 **EDGEDETECT:** Edge Detection Type Control bit
  - 1 = Detects any edge on the pin (CNx is used for the CN event)
  - 0 = Detects any edge on the pin (CNSTATx is used for the CN event)
- bit 10 **Reserved:** Always write '0'
- bit 9-0 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

## 11.0 TIMER1

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 14. “Timers”** (DS60001105), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

PIC32MK GPG/MCJ with CAN FD Family of devices feature one synchronous/asynchronous 16-bit timer that can operate as a free-running interval timer for various timing applications and counting external events. This timer can also be used with the low-power Secondary Oscillator (SOSC) for real-time clock applications.

The following modes are supported by Timer1:

- Synchronous Internal Timer
- Synchronous Internal Gated Timer
- Synchronous External Timer
- Asynchronous External Timer

### 11.1 Additional Supported Features

- Selectable clock prescaler
- Timer operation during Sleep and Idle modes
- Fast bit manipulation using CLR, SET, and INV registers
- Asynchronous mode can be used with the SOSC to function as a real-time clock
- ADC event trigger

### 11.2 Timer1 Usage Model Guidelines

#### 11.2.1 EXTERNAL CLOCK MODE OPERATION

When the Timer is operating with an external clock mode with the TCS bit (TxCON<1>) = 1, the mode bits of the TxCON register must be initialized using a separate Write operation from that used to enable the Timer. Specifically, the TCS, TSYNC, etc. bits must be written first, and then the ON bit (TxCON<15>) must be set in a subsequent write.

Once the ON bit is set, any writes to the TxCON register may cause erroneous counter operation.

**Note:** The ON bit should be clear when updates are made to any other bits in the TxCON register.

#### 11.2.2 ASYNCHRONOUS MODE OPERATION

When writing the ON bit when the Timer is configured in Asynchronous mode or in an external clock mode with the prescaler enabled, the act of setting the ON bit does not take effect until two rising edges of the external clock input have occurred.

#### 11.2.3 ASYNCHRONOUS MODE OPERATION WITH A PENDING TMRx REGISTER WRITE

When the Timer is configured in Asynchronous mode and the Timer is attempting to write to the TMRx register while a previous write is awaiting synchronization, the value written to the timer can become corrupted.

To ensure that writes will not cause the TMRx value to become corrupted, the TWDIS bit (TxCON<12>), when set, will ignore a write to the TMRx register when a previous write to the TMRx register is awaiting synchronization into the Asynchronous Timer Clock domain.

The TWIP bit (TxCON<11>) indicates when write synchronization is complete, and it is safe to write another value to the timer.

#### 11.2.4 PRx REGISTER WRITES

Writing to the PRx register while the Timer is active, may cause erratic operation.

#### 11.2.5 TIMER1 FORMULA

$$PR1 = ((Desired\ Time * FPBCLK2) - Prescaler\ Count)$$

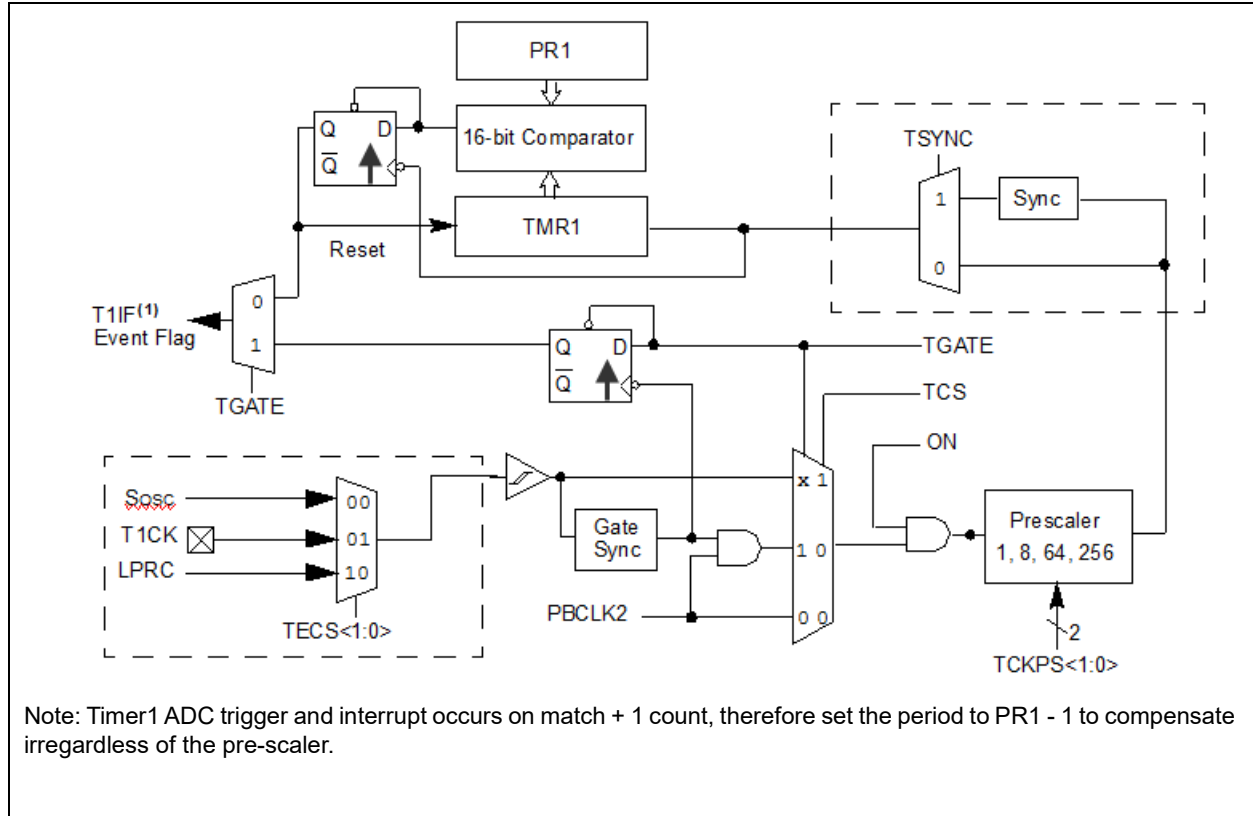
- Prescaler Count = 1, 8, 64, 256
- $FPBCLK2 = (FSYSCLK / PB2DIV<PBDIV>)$

**Note:** Timer1 interrupt occurs one additional prescaler cycle count *after* the timer and period match (TMR1 = PR1).



# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 11-1: TIMER1 BLOCK DIAGRAM



## 11.3 Timer1 Control Register

**TABLE 11-1: TIMER1 REGISTER MAP**

| Virtual Address<br>(BF82_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits       |       |       |       |       |       |           |       |      |            |      |       |      |      |      | All Resets |
|-----------------------------|---------------------------------|-----------|------------|-------|-------|-------|-------|-------|-----------|-------|------|------------|------|-------|------|------|------|------------|
|                             |                                 |           | 31/15      | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9      | 24/8  | 23/7 | 22/6       | 21/5 | 20/4  | 19/3 | 18/2 | 17/1 |            |
| 0000                        | T1CON                           | 31:16     | —          | —     | —     | —     | —     | —     | —         | —     | —    | —          | —    | —     | —    | —    | —    | 0000       |
|                             |                                 | 15:0      | ON         | —     | SIDL  | TWDIS | TWIP  | —     | TECS<1:0> | TGATE | —    | TCKPS<1:0> | —    | TSYNC | TCS  | —    | —    | 0000       |
| 0010                        | TMR1                            | 31:16     | —          | —     | —     | —     | —     | —     | —         | —     | —    | —          | —    | —     | —    | —    | —    | 0000       |
|                             |                                 | 15:0      | TMR1<15:0> |       |       |       |       |       |           |       |      |            |      |       |      |      |      | 0000       |
| 0020                        | PR1                             | 31:16     | —          | —     | —     | —     | —     | —     | —         | —     | —    | —          | —    | —     | —    | —    | —    | 0000       |
|                             |                                 | 15:0      | PR1<15:0>  |       |       |       |       |       |           |       |      |            |      |       |      |      |      | FFFF       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 11-1: T1CON: TYPE A TIMER CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | U-0            | R/W-0          | R/W-0          | R-0            | U-0            | R/W-0         | R/W-0         |
|           | ON             | —              | SIDL           | TWDIS          | TWIP           | —              | TECS<1:0>     |               |
| 7:0       | R/W-0          | U-0            | R/W-0          | R/W-0          | U-0            | R/W-0          | R/W-0         | U-0           |
|           | TGATE          | —              | TCKPS<1:0>     |                | —              | TSYNC          | TCS           | —             |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

- bit 31-16 **Unimplemented:** Read as '0'
- bit 15 **ON:** Timer On bit
  - 1 = Timer is enabled
  - 0 = Timer is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **SIDL:** Stop in Idle Mode bit
  - 1 = Discontinue operation when device enters Idle mode
  - 0 = Continue operation even in Idle mode
- bit 12 **TWDIS:** Asynchronous Timer Write Disable bit
  - 1 = Writes to TMR1 are ignored until pending write operation completes
  - 0 = Back-to-back writes are enabled (Legacy Asynchronous Timer functionality)
- bit 11 **TWIP:** Asynchronous Timer Write in Progress bit
  - In Asynchronous Timer mode:
  - 1 = Asynchronous write to TMR1 register in progress
  - 0 = Asynchronous write to TMR1 register complete
  - In Synchronous Timer mode:
  - This bit is read as '0'.
- bit 10 **Unimplemented:** Read as '0'
- bit 9-8 **TECS<1:0>:** Timer1 External Clock Selection bits
  - 11 =Reserved
  - 10 =External clock comes from the LPRC
  - 01 =External clock comes from the T1CK pin
  - 00 = External clock comes from the Sosc
- bit 7 **TGATE:** Timer Gated Time Accumulation Enable bit
  - When TCS = 1:
  - This bit is ignored.
  - When TCS = 0:
  - 1 = Gated time accumulation is enabled
  - 0 = Gated time accumulation is disabled
- bit 6 **Unimplemented:** Read as '0'
- bit 5-4 **TCKPS<1:0>:** Timer Input Clock Prescale Select bits
  - 11 =1:256 prescale value
  - 10 =1:64 prescale value
  - 01 =1:8 prescale value
  - 00 =1:1 prescale value

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 11-1: T1CON: TYPE A TIMER CONTROL REGISTER (CONTINUED)

- bit 3     **Unimplemented:** Read as '0'
- bit 2     **TSYNC:** Timer External Clock Input Synchronization Selection bit  
          When TCS = 1:  
          1 = External clock input is synchronized  
          0 = External clock input is not synchronized  
          When TCS = 0:  
          This bit is ignored.
- bit 1     **TCS:** Timer Clock Source Select bit  
          1 = External clock is defined by the TECS<1:0> bits  
          0 = Internal peripheral clock
- bit 0     **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 12.0 TIMER2 THROUGH TIMER9

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 14. “Timers”** (DS60001105), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The PIC32MK GPG/MCJ with CAN FD Family of devices features eight native synchronous/asynchronous 16/32-bit timers (default 16-bit mode) that can operate as free-running interval timers for various timing applications and counting external events.

## 12.1 Features

The following are key features of the timers:

- External 16/32-bit Counter Input mode
- Asynchronous external clock with/without selectable prescaler
- Synchronous internal clock with/without selectable prescaler
- External gate control (External pulse width measurement)
- Automatic timer synchronization control
- Operation in Idle mode
- Interrupt on a period register match or falling edge of external gate signal
- Time base for Input Capture and/or Output Compare modules

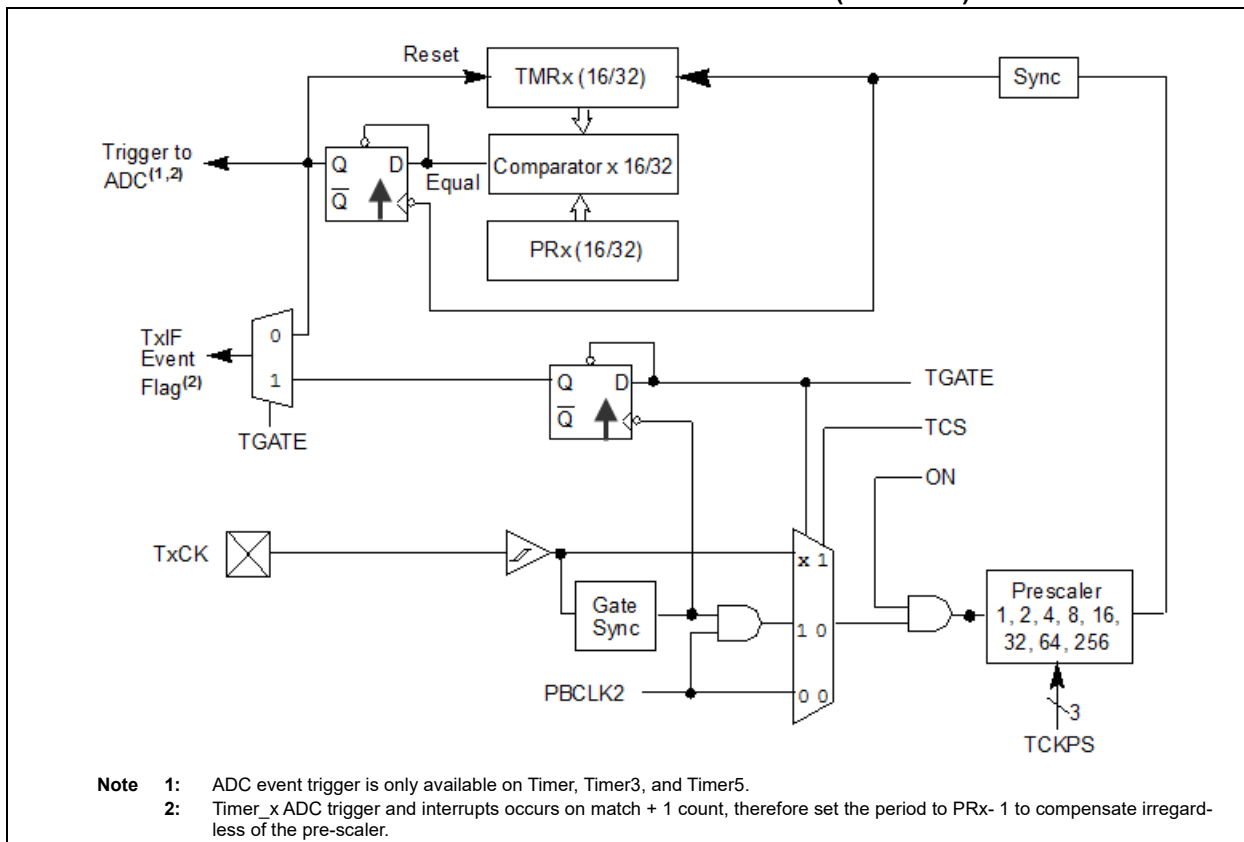
## 12.2 Timer2-Timer9 Formula

$$PRx = ((Desired\ Time * FPBCLK2) - Prescaler\ Count)$$

- Prescaler Count = 1, 2, 4, 8, 16, 32, 64, 256
- $FPBCLK2 = (FSYSCLK / PB2DIV < PBDIV >)$

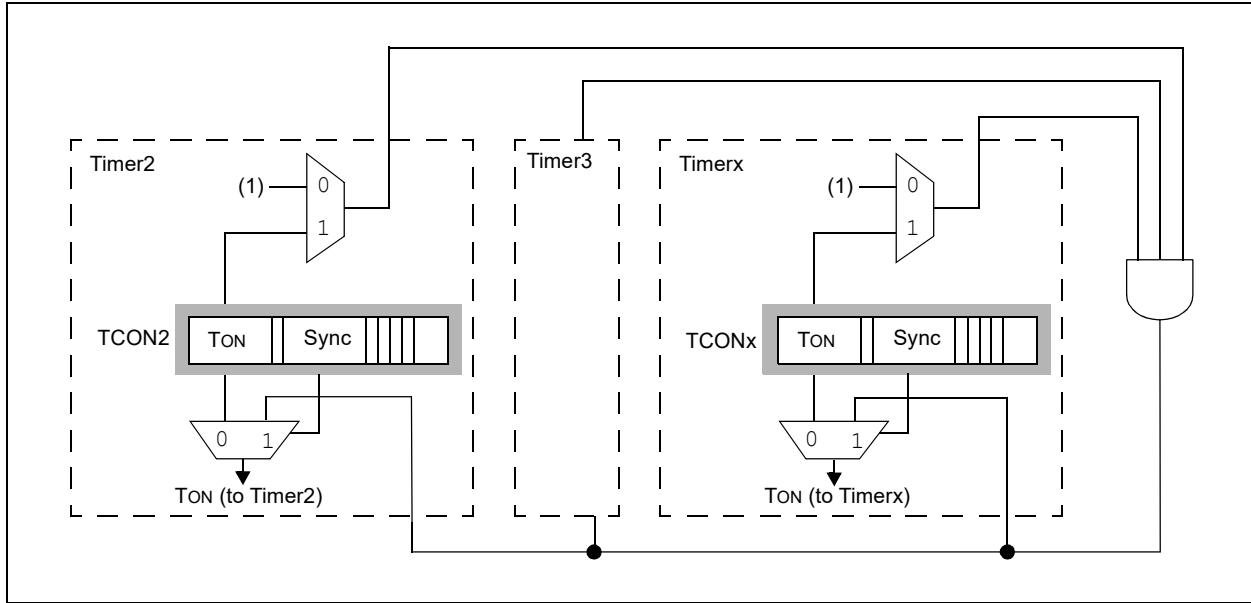
**Note:** Timer interrupts and triggers occur one additional prescaler cycle count *after* the timer and period match ( $TMRx = PRx$ ).

**FIGURE 12-1: TIMER2 THROUGH TIMER9 BLOCK DIAGRAM (16/32-BIT)**



# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 12-2: TIMER SYNCHRONIZATION BLOCK DIAGRAM



## 12.3 Timer2-Timer9 Control Registers

**TABLE 12-1: TIMER2 THROUGH TIMER9 REGISTER MAP**

| Virtual Address<br>(BF82_#) | Register<br>Name(s) | Bit Range | Bits        |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | All Resets |
|-----------------------------|---------------------|-----------|-------------|-------|-------|-------|-------|-------|------|------|------|-------|------------|------|------|------|------|------|------------|
|                             |                     |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6  | 21/5       | 20/4 | 19/3 | 18/2 | 17/1 | 16/0 |            |
| 0200                        | T2CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |
| 0210                        | TMR2                | 31:16     | TMR2<31:16> |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | TMR2<15:0>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
| 0220                        | PR2                 | 31:16     | PR2<31:16>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
|                             |                     | 15:0      | PR2<15:0>   |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
| 0400                        | T3CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |
| 0410                        | TMR3                | 31:16     | TMR3<31:16> |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | TMR3<15:0>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
| 0420                        | PR3                 | 31:16     | PR3<31:16>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
|                             |                     | 15:0      | PR3<15:0>   |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
| 0600                        | T4CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |
| 0610                        | TMR4                | 31:16     | TMR4<31:16> |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | TMR4<15:0>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
| 0620                        | PR4                 | 31:16     | PR4<31:16>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
|                             |                     | 15:0      | PR4<15:0>   |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
| 0800                        | T5CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |
| 0810                        | TMR5                | 31:16     | TMR5<31:16> |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | TMR5<15:0>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
| 0820                        | PR5                 | 31:16     | PR5<31:16>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
|                             |                     | 15:0      | PR5<15:0>   |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
| 0A00                        | T6CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |
| 0A10                        | TMR6                | 31:16     | TMR6<31:16> |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | TMR6<15:0>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | 0000       |
| 0A20                        | PR6                 | 31:16     | PR6<31:16>  |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
|                             |                     | 15:0      | PR6<15:0>   |       |       |       |       |       |      |      |      |       |            |      |      |      |      |      | FFFF       |
| 0C00                        | T7CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —    | —    | —    | —    | —    | 0000       |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |      |      | T32  | —    | TCS  | —          |

**Legend:** × = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.



TABLE 12-1: TIMER2 THROUGH TIMER9 REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register<br>Name(1) | Bit Range | Bits        |       |       |       |       |       |      |      |      |       |            |            |      |      | All Resets |      |      |
|-----------------------------|---------------------|-----------|-------------|-------|-------|-------|-------|-------|------|------|------|-------|------------|------------|------|------|------------|------|------|
|                             |                     |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6  | 21/5       | 20/4       | 19/3 | 18/2 |            | 17/1 | 16/0 |
| 0C10                        | TMR7                | 31:16     | TMR7<31:16> |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
|                             |                     | 15:0      | TMR7<15:0>  |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
| 0C20                        | PR7                 | 31:16     | PR7<31:16>  |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |
|                             |                     | 15:0      | PR7<15:0>   |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |
| 0E00                        | T8CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —          | —    | —    | —          | 0000 |      |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | SYNC | TGATE | TCKPS<2:0> |            | T32  | —    | TCS        | —    | 0000 |
| 0E10                        | TMR8                | 31:16     | TMR8<31:16> |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
|                             |                     | 15:0      | TMR8<15:0>  |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
| 0E20                        | PR8                 | 31:16     | PR8<31:16>  |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |
|                             |                     | 15:0      | PR8<15:0>   |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |
| 1000                        | T9CON               | 31:16     | —           | —     | —     | —     | —     | —     | —    | —    | —    | —     | —          | —          | —    | —    | —          | —    | 0000 |
|                             |                     | 15:0      | ON          | —     | SIDL  | —     | —     | —     | —    | —    | —    | SYNC  | TGATE      | TCKPS<2:0> |      | T32  | —          | TCS  | —    |
| 1010                        | TMR9                | 31:16     | TMR9<31:16> |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
|                             |                     | 15:0      | TMR9<15:0>  |       |       |       |       |       |      |      |      |       |            |            |      |      | 0000       |      |      |
| 1020                        | PR9                 | 31:16     | PR9<31:16>  |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |
|                             |                     | 15:0      | PR9<15:0>   |       |       |       |       |       |      |      |      |       |            |            |      |      | FFFF       |      |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 12-1: TxCON: TYPE B TIMER CONTROL REGISTER ('x' = 2-9)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 23:16     | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 15:8      | R/W-0<br>ON    | U-0<br>—       | R/W-0<br>SIDL  | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—      | R/W-0<br>SYNC |
| 7:0       | R/W-0<br>TGATE | R/W-0          | R/W-0          | R/W-0          | R/W-0          | U-0            | R/W-0         | U-0<br>—      |
|           |                | TCKPS<2:0>     |                |                | T32            | —              | TCS           | —             |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** Timer On bit  
 1 = Module is enabled  
 0 = Module is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Stop in Idle Mode bit  
 1 = Discontinue operation when device enters Idle mode  
 0 = Continue operation even in Idle mode

bit 12-9 **Unimplemented:** Read as '0'

bit 8 **SYNC:** TMRx Synchronized Timer Start/Stop Enable bit  
 1 = TMRx synchronized timer start/stop is enabled  
 0 = TMRx synchronized timer start/stop is disabled

**Note:** Setting this bit chains all timers whose corresponding SYNC bit is also set such that when the TON bit of all corresponding timers is set, the timers are enabled simultaneously. If any timers in the group are disabled, they are all disabled simultaneously. See [Figure 12-2](#) for additional information.

bit 7 **TGATE:** Timer Gated Time Accumulation Enable bit

When TCS = 1:  
 This bit is ignored and is read as '0'.

When TCS = 0:  
 1 = Gated time accumulation is enabled  
 0 = Gated time accumulation is disabled

bit 6-4 **TCKPS<2:0>:** Timer Input Clock Prescale Select bits

111 = 1:256 prescale value  
 110 = 1:64 prescale value  
 101 = 1:32 prescale value  
 100 = 1:16 prescale value  
 011 = 1:8 prescale value  
 010 = 1:4 prescale value  
 001 = 1:2 prescale value  
 000 = 1:1 prescale value

bit 3 **T32:** 32-Bit Timer Mode Select bit

1 = 32-bit Timer mode  
 0 = 16-bit Timer mode

bit 2 **Unimplemented:** Read as '0'

bit 1 **TCS:** Timer Clock Source Select bit

1 = External clock from TxCK pin  
 0 = Internal peripheral clock

bit 0 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 13.0 DEADMAN TIMER (DMT)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 9. “Watchdog, Deadman, and Power-up Timers”** (DS60001114), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

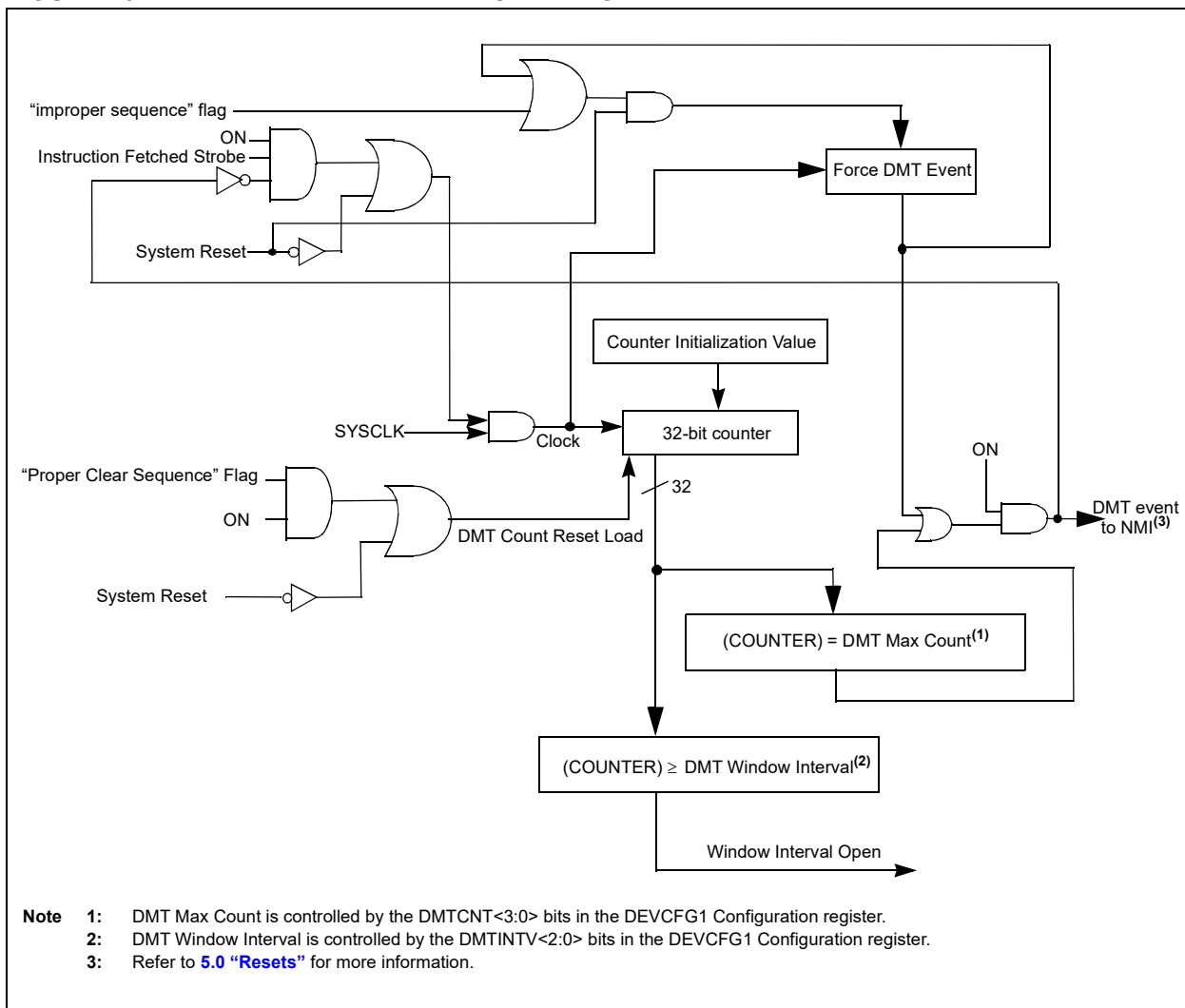
The primary function of the Deadman Timer (DMT) is to reset the processor in the event of a software malfunction. The DMT is a free-running instruction fetch timer, which is clocked whenever an instruction fetch occurs until a count match occurs. Instructions are not fetched when the processor is in Sleep mode.

The DMT consists of a 32-bit counter with a time-out count match value as specified by the DMTCNT<3:0> bits in the DEVCFG1 Configuration register.

A Deadman Timer is typically used in mission critical and safety critical applications, where any single failure of the software functionality and sequencing must be detected.

Figure 13-1 shows a block diagram of the Deadman Timer module.

**FIGURE 13-1: DEADMAN TIMER BLOCK DIAGRAM**



13.1 Deadman Timer Control Registers

TABLE 13-1: DEADMAN TIMER REGISTER MAP

| Virtual Address<br>(BF80_#) | Register Name | Bit Range | Bits          |       |       |       |       |       |      |      |            |      |          |      |      |      |      | All Resets |      |
|-----------------------------|---------------|-----------|---------------|-------|-------|-------|-------|-------|------|------|------------|------|----------|------|------|------|------|------------|------|
|                             |               |           | 31/15         | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7       | 22/6 | 21/5     | 20/4 | 19/3 | 18/2 | 17/1 |            | 16/0 |
| 0E00                        | DMTCON        | 31:16     | —             | —     | —     | —     | —     | —     | —    | —    | —          | —    | —        | —    | —    | —    | —    | —          | 0000 |
|                             |               | 15:0      | ON            | —     | —     | —     | —     | —     | —    | —    | —          | —    | —        | —    | —    | —    | —    | —          | 0000 |
| 0E10                        | DMTPRECLR     | 31:16     | —             | —     | —     | —     | —     | —     | —    | —    | —          | —    | —        | —    | —    | —    | —    | 0000       |      |
|                             |               | 15:0      | STEP1<7:0>    |       |       |       |       |       |      |      |            | —    | —        | —    | —    | —    | —    | —          | 0000 |
| 0E20                        | DMCLR         | 31:16     | —             | —     | —     | —     | —     | —     | —    | —    | —          | —    | —        | —    | —    | —    | —    | 0000       |      |
|                             |               | 15:0      | —             | —     | —     | —     | —     | —     | —    | —    | STEP2<7:0> |      |          |      |      |      |      | 0000       |      |
| 0E30                        | DMTSTAT       | 31:16     | —             | —     | —     | —     | —     | —     | —    | —    | —          | —    | —        | —    | —    | —    | —    | 0000       |      |
|                             |               | 15:0      | —             | —     | —     | —     | —     | —     | —    | —    | BAD1       | BAD2 | DMTEVENT | —    | —    | —    | —    | WINOPN     | 0000 |
| 0E40                        | DMTCNT        | 31:16     | COUNTER<31:0> |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |
|                             |               | 15:0      | COUNTER<31:0> |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |
| 0E60                        | DMTPSCNT      | 31:16     | PSCNT<31:0>   |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |
|                             |               | 15:0      | PSCNT<31:0>   |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |
| 0E70                        | DMTPSINTV     | 31:16     | PSINTV<31:0>  |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |
|                             |               | 15:0      | PSINTV<31:0>  |       |       |       |       |       |      |      |            |      |          |      |      |      |      | 0000       |      |

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 13-1: DMTCON: DEADMAN TIMER CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7    | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0               | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                 | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0               | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                 | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0             | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | ON <sup>(1)</sup> | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | U-0               | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                 | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-16     **Unimplemented:** Read as '0'  
 bit 15        **ON:** Deadman Timer Module Enable bit<sup>(1)</sup>  
               1 = Deadman Timer module is enabled  
               0 = Deadman Timer module is disabled  
 bit 13-0     **Unimplemented:** Read as '0'

**Note 1:** This bit only has control when FDMTEN (DEVCFG1<3>) = 0.

**REGISTER 13-2: DMTPRECLR: DEADMAN TIMER PRECLEAR REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | STEP1<7:0>     |                |                |                |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-16     **Unimplemented:** Read as '0'  
 bit 15-8     **STEP1<7:0>:** Preclear Enable bits  
               01000000 = Enables the Deadman Timer Preclear (Step 1)  
               All other write patterns = Set BAD1 flag.  
               These bits are cleared when a DMT reset event occurs. STEP1<7:0> is also cleared if the  
               STEP2<7:0> bits are loaded with the correct value in the correct sequence.  
 bit 7-0       **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 13-3: DMTCLR: DEADMAN TIMER CLEAR REGISTER**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | STEP2<7:0>        |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-8      **Unimplemented:** Read as '0'

bit 7-0      **STEP2<7:0>:** Clear Timer bits

00001000 = Clears STEP1<7:0>, STEP2<7:0> and the Deadman Timer if, and only if, preceded by correct loading of STEP1<7:0> bits in the correct sequence. The write to these bits may be verified by reading DMTCNT and observing the counter being reset.

All other write patterns = Set BAD2 bit, the value of STEP1<7:0> will remain unchanged, and the new value being written STEP2<7:0> will be captured. These bits are also cleared when a DMT reset event occurs.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 13-4: DMTSTAT: DEADMAN TIMER STATUS REGISTER**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 7:0       | R-0, HC           | R-0, HC           | R-0, HC           | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R-0              |
|           | BAD1              | BAD2              | DMTEVENT          |                   |                   |                   |                  | WINOPN           |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | HC = Hardware Cleared              |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | '1' = Bit is set                   |
|                   | U = Unimplemented bit, read as '0' |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

- bit 31-8     **Unimplemented:** Read as '0'
- bit 7       **BAD1:** Bad STEP1<7:0> Value Detect bit  
           1 = Incorrect STEP1<7:0> value or out of sequence write to STEP2<7:0> was detected  
           0 = Incorrect STEP1<7:0> value was not detected
- bit 6       **BAD2:** Bad STEP2<7:0> Value Detect bit  
           1 = Incorrect STEP2<7:0> value was detected  
           0 = Incorrect STEP2<7:0> value was not detected
- bit 5       **DMTEVENT:** Deadman Timer Event bit  
           1 = Deadman timer event was detected (counter expired or bad STEP1<7:0> or STEP2<7:0> value was entered prior to counter increment)  
           0 = Deadman timer even was not detected  
           **Note:** This bit is cleared only on a Reset.
- bit 4-1     **Unimplemented:** Read as '0'
- bit 0       **WINOPN:** Deadman Timer Clear Window bit  
           1 = Deadman timer clear window is open  
           0 = Deadman timer clear window is not open





# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 13-7: DMTPSINTV: POST STATUS CONFIGURE DMT INTERVAL STATUS REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | PSINTV<31:24>     |                   |                   |                   |                   |                   |                  |                  |
| 23:16     | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | PSINTV<23:16>     |                   |                   |                   |                   |                   |                  |                  |
| 15:8      | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | PSINTV<15:8>      |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R-0               | R-0               | R-0               | R-0               | R-0               | R-y               | R-y              | R-y              |
|           | PSINTV<7:0>       |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

y= Value set from Configuration bits on POR  
U = Unimplemented bit, read as '0'  
'0' = Bit is cleared      x = Bit is unknown

bit 31-8

**PSINTV<31:0>**: DMT Window Interval Configuration Status bits

This is always the value of the DMTINTV<2:0> bits in the DEVCFG1 Configuration register.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ WITH CAN FD FAMILY

## 14.0 WATCHDOG TIMER (WDT)

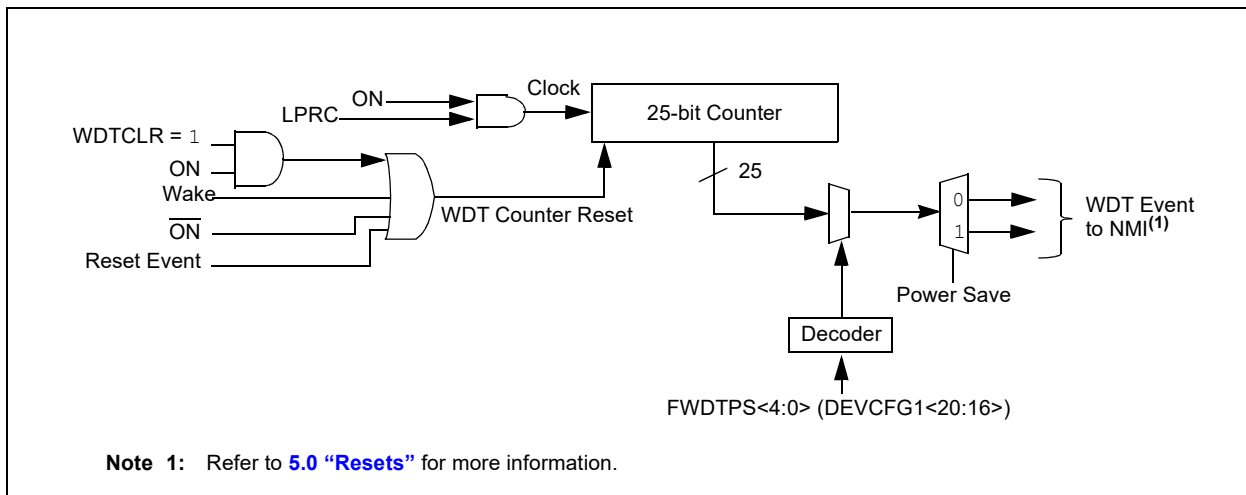
**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 9. “Watchdog, Deadman, and Power-up Timers”** (DS60001114), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

When enabled, the Watchdog Timer (WDT) operates from the internal Low-Power Oscillator (LPRC) clock source and can be used to detect system software malfunctions by resetting the device if the WDT is not cleared periodically in software. Various WDT time-out periods can be selected using the WDT postscaler. The WDT can also be used to wake the device from Sleep or Idle mode.

Some of the key features of the WDT module are:

- Configuration or software controlled
- User-configurable time-out period
- Can wake the device from Sleep or Idle

**FIGURE 14-1: WATCHDOG TIMER BLOCK DIAGRAM**



14.1 Watchdog Timer Control Registers

TABLE 14-1: WATCHDOG TIMER REGISTER MAP

| Virtual Address<br>(BF80_#) | Register Name         | Bit Range | Bits            |       |       |             |       |       |      |      |      |             |      |      |      |          | All Resets |
|-----------------------------|-----------------------|-----------|-----------------|-------|-------|-------------|-------|-------|------|------|------|-------------|------|------|------|----------|------------|
|                             |                       |           | 31/15           | 30/14 | 29/13 | 28/12       | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6        | 21/5 | 20/4 | 19/3 | 18/2     |            |
| 0C00                        | WDTCON <sup>(1)</sup> | 31:16     | WDTCLRKEY<15:0> |       |       |             |       |       |      |      |      |             |      |      |      |          | 0000       |
|                             |                       | 15:0      | ON              | —     | —     | RUNDIV<4:0> |       |       |      | —    | —    | SLPDIV<4:0> |      |      |      | WDTWINEN | 0000       |

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: This register has corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See 13.2 "CLR, SET, and INV Registers" for more information.

# PIC32MK GPG/MCJ WITH CAN FD FAMILY

## REGISTER 14-1: WDTCON: WATCHDOG TIMER CONTROL REGISTER

| Bit Range       | Bit 31/23/15/7    | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------------|-------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24           | W-0               | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
| WDTCLRKEY<15:8> |                   |                |                |                |                |                |               |               |
| 23:16           | W-0               | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
| WDTCLRKEY<7:0>  |                   |                |                |                |                |                |               |               |
| 15:8            | R/W-0             | U-0            | U-0            | R-y            | R-y            | R-y            | R-y           | R-y           |
|                 | ON <sup>(1)</sup> | —              | —              | RUNDIV<4:0>    |                |                |               |               |
| 7:0             | U-0               | U-0            | R-y            | R-y            | R-y            | R-y            | R-y           | R/W-0         |
|                 | —                 | —              | SLPDIV<4:0>    |                |                |                |               | WDTWINEN      |

|                   |  |
|-------------------|--|
| <b>Legend:</b>    | y = Values set from Configuration bits on POR                      |
| R = Readable bit  | W = Writable bit      U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **WDTCLRKEY<15:0>**: Watchdog Timer Clear Key bits

To clear the Watchdog Timer to prevent a time-out, software must write the value 0x5743 to these bits using a single 16-bit write.

bit 15 **ON**: Watchdog Timer Enable bit<sup>(1)</sup>

1 = The Watchdog Timer module is enabled  
0 = The Watchdog Timer module is disabled

bit 14-13 **Unimplemented**: Read as '0'

bit 12-8 **RUNDIV<4:0>**: Watchdog Timer Postscaler Value in Run Mode bits

In Run mode, these bits are set to the values of the WDTPS<4:0> Configuration bits in the DEVCFG1 register.

bit 7-6 **Unimplemented**: Read as '0'

bit 5-1 **SLPDIV<4:0>**: Watchdog Timer Postscaler Value in Sleep Mode bits

In Sleep mode, these bits are set to the values of the WDTPS <4:0> Configuration bits in the DEVCFG1 register.

bit 0 **WDTWINEN**: Watchdog Timer Window Enable bit

1 = Enable windowed Watchdog Timer  
0 = Disable windowed Watchdog Timer

**Note 1:** This bit only has control when FWDTEN (DEVCFG1<23>) = 0.

# PIC32MK GPG/MCJ WITH CAN FD FAMILY

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 15.0 INPUT CAPTURE

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 15. “Input Capture”** (DS60001122), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Input Capture module is useful in applications requiring frequency (period) and pulse measurement.

The Input Capture module captures the 16-bit or 32-bit value of the selected Time Base registers when an event occurs at the ICx pin.

Capture events are caused by the following factors:

- Capture timer value on every edge (rising and falling), specified edge first
- Prescaler capture event modes:
  - Capture every falling edge of input at ICx pin
  - Capture every rising edge of input at ICx pin

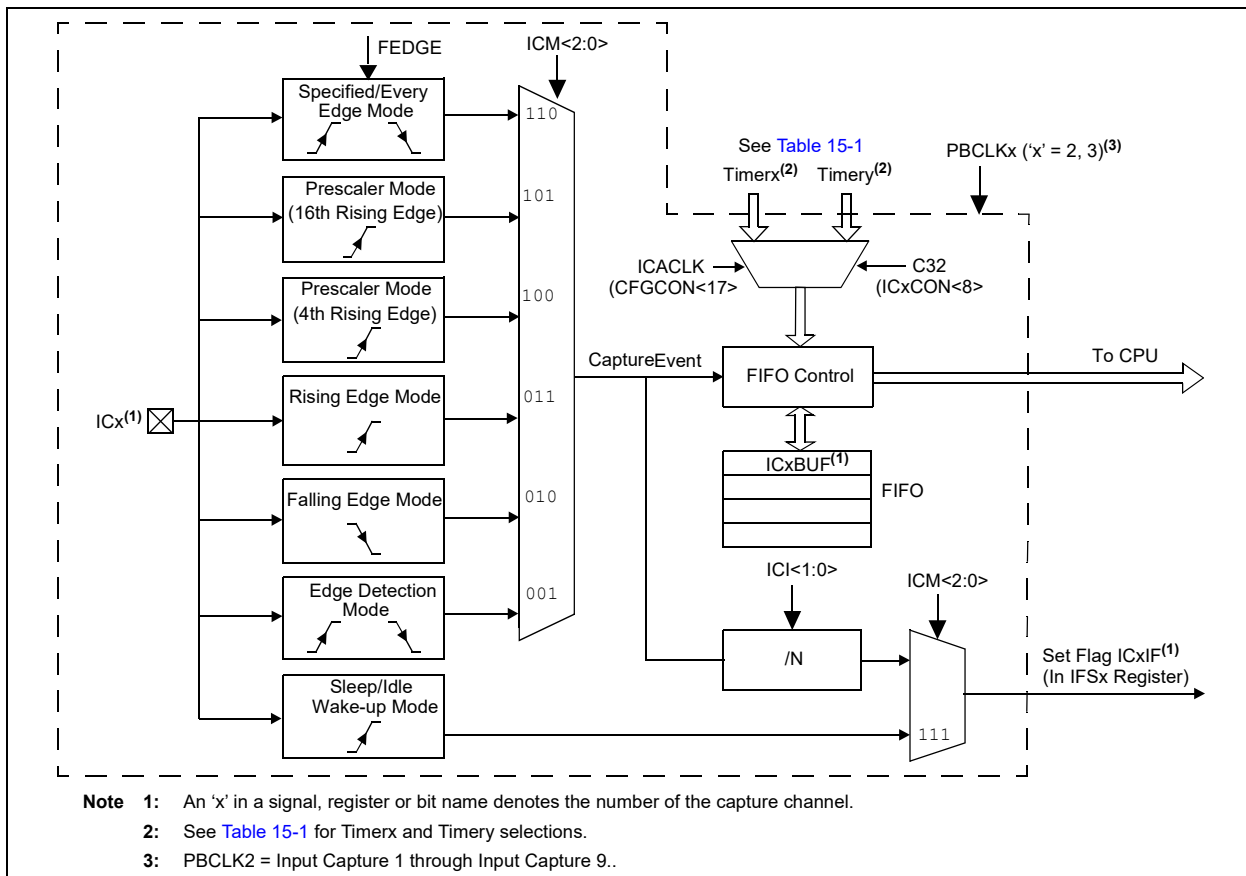
- Capture every 4th rising edge of input at ICx pin
- Capture every 16th rising edge of input at ICx pin
- Capture every rising and falling edge of input at ICx pin
- Capture timer values based on internal or external clocks

Each input capture channel can select between either seven 16-bit time bases or three 32-bit time base. The selected timer can use either an internal or external clock.

Other operational features include:

- Device wake-up from capture pin during Sleep and Idle modes
- Interrupt on input capture event
- 4-word FIFO buffer for capture values; Interrupt optionally generated after 1, 2, 3, or 4 buffer locations are filled
- Input capture can also be used to provide additional sources of external interrupts
- sources of external interrupts

**FIGURE 15-1: INPUT CAPTURE BLOCK DIAGRAM**





# PIC32MK GPG/MCJ with CAN FD Family

The timer source for each Input Capture module depends on the setting of the ICACLK bit in the CFGCON register and the C32 bit in the ICxCON register. The available configurations are shown in [Table 15-1](#).

**TABLE 15-1: TIMER SOURCE CONFIGURATIONS**

| ICx      | ICACLK<br>(CFGCON<17>) | C32<br>ICxCON<8> | ICTMR<br>ICxCON<7> | Timerx     | Timery     | ICxBUF Contents |
|----------|------------------------|------------------|--------------------|------------|------------|-----------------|
| IC1-IC3  | 0                      | 0                | 0                  | —          | TMR3<15:0> | TMR3<15:0>      |
|          |                        |                  | 1                  | TMR2<15:0> | —          | TMR2<15:0>      |
|          | 0                      | 1                | x                  | TMR2<31:0> | TMR2<31:0> | TMR2<31:0>      |
|          | 1                      | 0                | 0                  | —          | TMR5<15:0> | TMR5<15:0>      |
|          |                        |                  | 1                  | TMR4<15:0> | —          | TMR4<15:0>      |
| 1        | 1                      | x                | TMR4<31:0>         | TMR4<31:0> | TMR4<31:0> |                 |
| IC4-IC6, | 0                      | 0                | 0                  | —          | TMR3<15:0> | TMR3<15:0>      |
|          |                        |                  | 1                  | TMR2<15:0> | —          | TMR2<15:0>      |
|          | 0                      | 1                | x                  | TMR2<31:0> | TMR2<31:0> | TMR2<31:0>      |
|          | 1                      | 0                | 0                  | —          | TMR3<15:0> | TMR3<15:0>      |
|          |                        |                  | 1                  | TMR2<15:0> | —          | TMR2<15:0>      |
| 1        | 1                      | x                | TMR2<31:0>         | TMR2<31:0> | TMR2<31:0> |                 |
| IC7-IC9  | 0                      | 0                | 0                  | —          | TMR3<15:0> | TMR3<15:0>      |
|          |                        |                  | 1                  | TMR2<15:0> | —          | TMR2<15:0>      |
|          | 0                      | 1                | x                  | TMR2<31:0> | TMR2<31:0> | TMR2 <31:0>     |
|          | 1                      | 0                | 0                  | —          | TMR7<15:0> | TMR7<15:0>      |
|          |                        |                  | 1                  | TMR6<15:0> | —          | TMR6<15:0>      |
| 1        | 1                      | x                | TMR6<31:0>         | TMR6<31:0> | TMR6<31:0> |                 |

## 15.1 Input Capture Control Registers

**TABLE 15-2: INPUT CAPTURE 1 THROUGH INPUT CAPTURE 9 REGISTER MAP**

| Virtual Address<br>BF82_# | Register<br>Name      | Bit Range | Bits         |       |       |       |       |       |       |      |       |          |      |       |          |      |      | All Resets |      |
|---------------------------|-----------------------|-----------|--------------|-------|-------|-------|-------|-------|-------|------|-------|----------|------|-------|----------|------|------|------------|------|
|                           |                       |           | 31/15        | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9  | 24/8 | 23/7  | 22/6     | 21/5 | 20/4  | 19/3     | 18/2 | 17/1 |            | 16/0 |
| 2000                      | IC1CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2010                      | IC1BUF                | 31:16     | IC1BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2200                      | IC2CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2210                      | IC2BUF                | 31:16     | IC2BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2400                      | IC3CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2410                      | IC3BUF                | 31:16     | IC3BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2600                      | IC4CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2610                      | IC4BUF                | 31:16     | IC4BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2800                      | IC5CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2810                      | IC5BUF                | 31:16     | IC5BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2A00                      | IC6CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2A10                      | IC6BUF                | 31:16     | IC6BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2C00                      | IC7CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2C10                      | IC7BUF                | 31:16     | IC7BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 2E00                      | IC8CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 2E10                      | IC8BUF                | 31:16     | IC8BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
| 3000                      | IC9CON <sup>(1)</sup> | 31:16     | —            | —     | —     | —     | —     | —     | —     | —    | —     | —        | —    | —     | —        | —    | —    | —          | 0000 |
|                           |                       | 15:0      | ON           | —     | SIDL  | —     | —     | —     | FEDGE | C32  | ICTMR | ICI<1:0> | ICOV | ICBNE | ICM<2:0> | 0000 |      |            |      |
| 3010                      | IC9BUF                | 31:16     | IC9BUF<31:0> |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |
|                           |                       | 15:0      |              |       |       |       |       |       |       |      |       |          |      |       |          |      |      | xxxx       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** This register has corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 15-1: ICxCON: INPUT CAPTURE 'x' CONTROL REGISTER ('x' = 1-9)**

| Bit Range | Bit 31/23/15/7       | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                  | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                    | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0                  | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                    | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0                | U-0            | R/W-0          | U-0            | U-0            | U-0            | R/W-0         | R/W-0         |
|           | ON                   | —              | SIDL           | —              | —              | —              | FEDGE         | C32           |
| 7:0       | R/W-0                | R/W-0          | R/W-0          | R-0            | R-0            | R/W-0          | R/W-0         | R/W-0         |
|           | ICTMR <sup>(1)</sup> | ICI<1:0>       |                | ICOV           | ICBNE          | ICM<2:0>       |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit

-n = Bit Value at POR: ('0', '1', x = unknown)

P = Programmable bit

r = Reserved bit

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** Input Capture Module Enable bit

1 = Module enabled

0 = Disable and reset module, disable clocks, disable interrupt generation and allow SFR modifications

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Stop in Idle Control bit

1 = Halt in CPU Idle mode

0 = Continue to operate in CPU Idle mode

bit 12-10 **Unimplemented:** Read as '0'

bit 9 **FEDGE:** First Capture Edge Select bit (only used in mode 6, ICM<2:0> = 110)

1 = Capture rising edge first

0 = Capture falling edge first

bit 8 **C32:** 32-bit Capture Select bit

1 = 32-bit timer resource capture

0 = 16-bit timer resource capture

bit 7 **ICTMR:** Timer Select bit (Does not affect timer selection when C32 (ICxCON<8>) is '1')<sup>(1)</sup>

0 = Timery is the counter source for capture

1 = Timerx is the counter source for capture

bit 6-5 **ICI<1:0>:** Interrupt Control bits

11 = Interrupt on every fourth capture event

10 = Interrupt on every third capture event

01 = Interrupt on every second capture event

00 = Interrupt on every capture event

bit 4 **ICOV:** Input Capture Overflow Status Flag bit (read-only)

1 = Input capture overflow occurred

0 = No input capture overflow occurred

bit 3 **ICBNE:** Input Capture Buffer Not Empty Status bit (read-only)

1 = Input capture buffer is not empty; at least one more capture value can be read

0 = Input capture buffer is empty

**Note 1:** Refer to [Table 15-1](#) for Timerx and Timery selections.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 15-1: ICxCON: INPUT CAPTURE 'x' CONTROL REGISTER ('x' = 1-9) (CONTINUED)

bit 2-0      **ICM<2:0>**: Input Capture Mode Select bits

- 111 = Interrupt-Only mode (only supported while in Sleep mode or Idle mode)
- 110 = Simple Capture Event mode – every edge, specified edge first and every edge thereafter
- 101 = Prescaled Capture Event mode – every sixteenth rising edge
- 100 = Prescaled Capture Event mode – every fourth rising edge
- 011 = Simple Capture Event mode – every rising edge
- 010 = Simple Capture Event mode – every falling edge
- 001 = Edge Detect mode – every edge (rising and falling)
- 000 = Input Capture module is disabled

**Note 1:** Refer to [Table 15-1](#) for Timerx and Timery selections.

# PIC32MK GPG/MCJ with CAN FD Family

## 16.0 OUTPUT COMPARE

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 16. “Output Compare”** (DS60001111), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Output Compare module is used to generate a single pulse or a train of pulses in response to selected time base events.

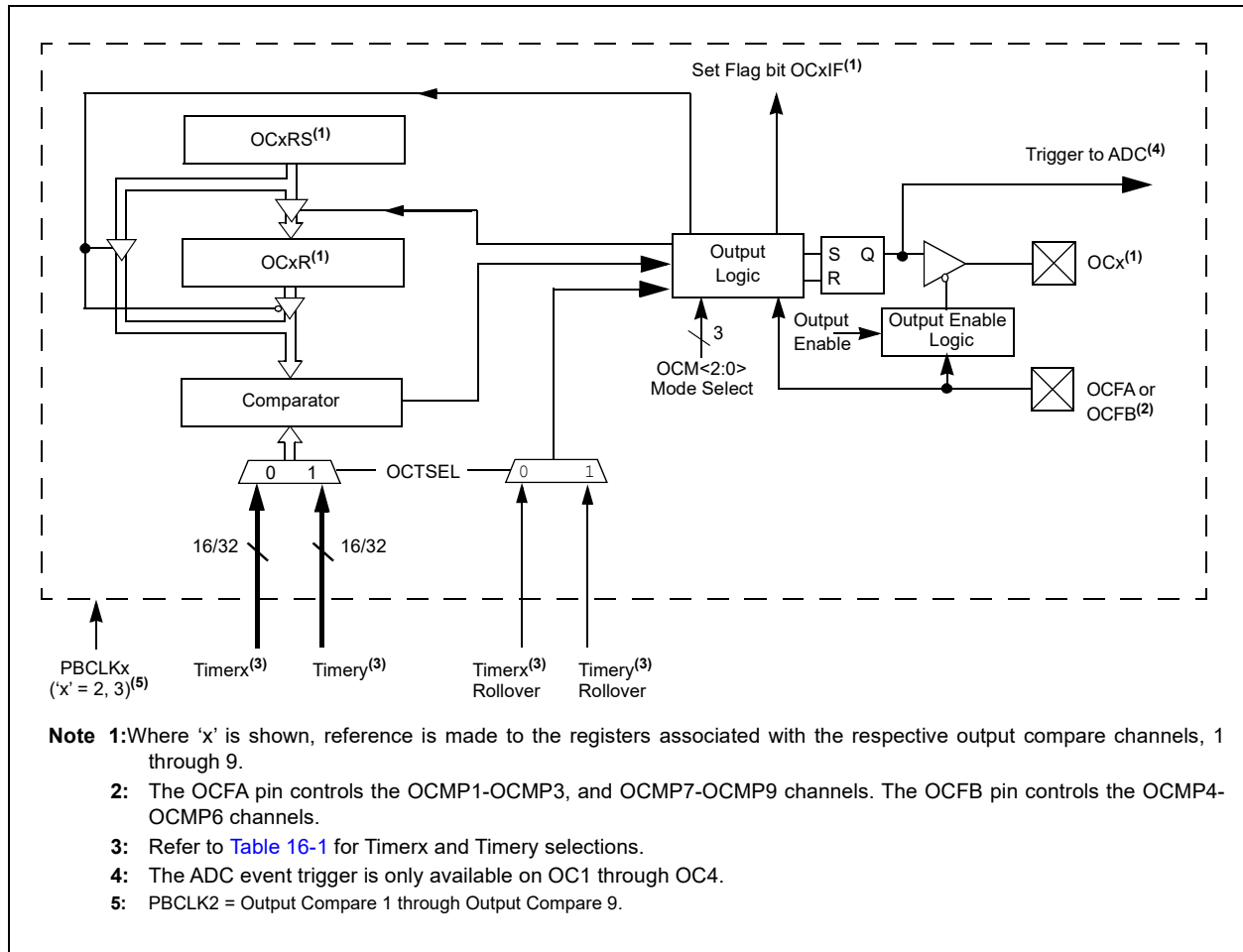
For all modes of operation, the Output Compare module compares the values stored in the OCxR and/or the OCxRS registers to the value in the selected timer.

When a match occurs, the Output Compare module generates an event based on the selected mode of operation.

The following are some of the key features of the Output Compare:

- Multiple Output Compare modules in a device
- Programmable interrupt generation on compare event
- Single and Dual Compare modes
- Single and continuous output pulse generation
- Pulse-Width Modulation (PWM) mode
- Hardware-based PWM Fault detection and automatic output disable
- Programmable selection of 16-bit or 32-bit time bases
- Can operate from either of two available 16-bit time bases or a single 32-bit time base
- ADC event trigger for OC1 through OC4

**FIGURE 16-1: OUTPUT COMPARE MODULE BLOCK DIAGRAM**



# PIC32MK GPG/MCJ with CAN FD Family

The timer source for each Output Compare module depends on the setting of the OCACLK bit in the CFGCON register, the OC32 bit in the OCxCON register, and the OCTSEL bit in the OCxCON register. The available configurations are shown in [Table 16-1](#).

| OC Module | Fault Input |
|-----------|-------------|
| OCMP1-3   | OCFA        |
| OCMP4-6   | OCFB        |
| OCMP7-9   | OCFA        |

**TABLE 16-1: TIMER SOURCE CONFIGURATIONS**

| OCx     | OCACLK<br>CFGCON<16> | OC32<br>(OCxCON<5> | OCTSEL<br>OCxCON<3> | Timerx     | Timery     | Output<br>Compare<br>Timer Source |            |
|---------|----------------------|--------------------|---------------------|------------|------------|-----------------------------------|------------|
| OC1-OC3 | 0                    | 0                  | 0                   | TMR2<15:0> | —          | TMR2<15:0>                        |            |
|         |                      |                    | 1                   | —          | TMR3<15:0> | TMR3<15:0>                        |            |
|         | 0                    | 1                  | 0                   | TMR2<31:0> | —          | TMR2<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR2<31:0> | TMR2<31:0>                        |            |
|         | 1                    | 0                  | 0                   | TMR4<15:0> | —          | TMR4<15:0>                        |            |
|         |                      |                    | 1                   | —          | TMR5<15:0> | TMR5<15:0>                        |            |
|         | 1                    | 1                  | 0                   | TMR4<31:0> | —          | TMR4<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR4<31:0> | TMR4<31:0>                        |            |
|         | OC4-OC6,             | 0                  | 0                   | 0          | TMR2<15:0> | —                                 | TMR2<15:0> |
|         |                      |                    |                     | 1          | —          | TMR3<15:0>                        | TMR3<15:0> |
| 0       |                      | 1                  | 0                   | TMR2<31:0> | —          | TMR2<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR2<31:0> | TMR2<31:0>                        |            |
| 1       |                      | 0                  | 0                   | TMR2<15:0> | —          | TMR2<15:0>                        |            |
|         |                      |                    | 1                   | —          | TMR3<15:0> | TMR3<15:0>                        |            |
| 1       |                      | 1                  | 0                   | TMR2<31:0> | —          | TMR2<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR2<31:0> | TMR2<31:0>                        |            |
| OC7-OC9 |                      | 0                  | 0                   | 0          | TMR2<15:0> | —                                 | TMR2<15:0> |
|         |                      |                    |                     | 1          | —          | TMR3<15:0>                        | TMR3<15:0> |
|         | 0                    | 1                  | 0                   | TMR2<31:0> | —          | TMR2<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR2<31:0> | TMR2<31:0>                        |            |
|         | 1                    | 0                  | 0                   | TMR6<15:0> | —          | TMR6<15:0>                        |            |
|         |                      |                    | 1                   | —          | TMR7<15:0> | TMR7<15:0>                        |            |
|         | 1                    | 1                  | 0                   | TMR6<31:0> | —          | TMR6<31:0>                        |            |
|         |                      |                    | 1                   | —          | TMR6<31:0> | TMR6<31:0>                        |            |

# PIC32MK GPG/MCJ with CAN FD Family

## 16.1 Output Compare Control Registers

**REGISTER 16-1: OCxCON: OUTPUT COMPARE 'x' CONTROL REGISTER ('x' = 1-16)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4       | Bit 27/19/11/3        | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------------|-----------------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0                  | U-0                   | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —                    | —                     | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0                  | U-0                   | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —                    | —                     | —              | —             | —             |
| 15:8      | R/W-0          | U-0            | R/W-0          | U-0                  | U-0                   | U-0            | U-0           | U-0           |
|           | ON             | —              | SIDL           | —                    | —                     | —              | —             | —             |
| 7:0       | U-0            | U-0            | R/W-0          | R-0, HS, HC          | R/W-0                 | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | OC32           | OCFLT <sup>(1)</sup> | OCTSEL <sup>(2)</sup> | OCM<2:0>       |               |               |

|                   |                      |                                    |
|-------------------|----------------------|------------------------------------|
| <b>Legend:</b>    | HS = Set in hardware | HC = Cleared by hardware           |
| R = Readable bit  | W = Writable bit     | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set     | '0' = Bit is cleared               |
|                   |                      | x = Bit is unknown                 |

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** Output Compare Peripheral On bit  
 1 = Output Compare peripheral is enabled  
 0 = Output Compare peripheral is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Stop in Idle Mode bit  
 1 = Discontinue operation when CPU enters Idle mode  
 0 = Continue operation in Idle mode

bit 12-6 **Unimplemented:** Read as '0'

bit 5 **OC32:** 32-bit Compare Mode bit  
 1 = OCxR<31:0> and/or OCxRS<31:0> are used for comparisons to the 32-bit timer source  
 0 = OCxR<15:0> and OCxRS<15:0> are used for comparisons to the 16-bit timer source

bit 4 **OCFLT:** PWM Fault Condition Status bit<sup>(1)</sup>  
 1 = PWM Fault condition has occurred (cleared in HW only)  
 0 = No PWM Fault condition has occurred

bit 3 **OCTSEL:** Output Compare Timer Select bit<sup>(2)</sup>  
 1 = Timery is the clock source for this Output Compare module  
 0 = Timerx is the clock source for this Output Compare module

bit 2-0 **OCM<2:0>:** Output Compare Mode Select bits  
 111 = PWM mode on OCx; Fault pin enabled  
 110 = PWM mode on OCx; Fault pin disabled  
 101 = Initialize OCx pin low; generate continuous output pulses on OCx pin  
 100 = Initialize OCx pin low; generate single output pulse on OCx pin  
 011 = Compare event toggles OCx pin  
 010 = Initialize OCx pin high; compare event forces OCx pin low  
 001 = Initialize OCx pin low; compare event forces OCx pin high  
 000 = Output compare peripheral is disabled but continues to draw current

**Note 1:** This bit is only used when OCM<2:0> = '111'. It is read as '0' in all other modes.

**Note 2:** Refer to [Table 16-1](#) for Timerx and Timery selections.

# PIC32MK GPG/MCJ with CAN FD Family

---

NOTES:



# PIC32MK GPG/MCJ with CAN FD Family

## 17.0 SERIAL PERIPHERAL INTERFACE (SPI) AND INTER-IC SOUND (I<sup>2</sup>S)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 23. “Serial Peripheral Interface (SPI)”** (DS60001106), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

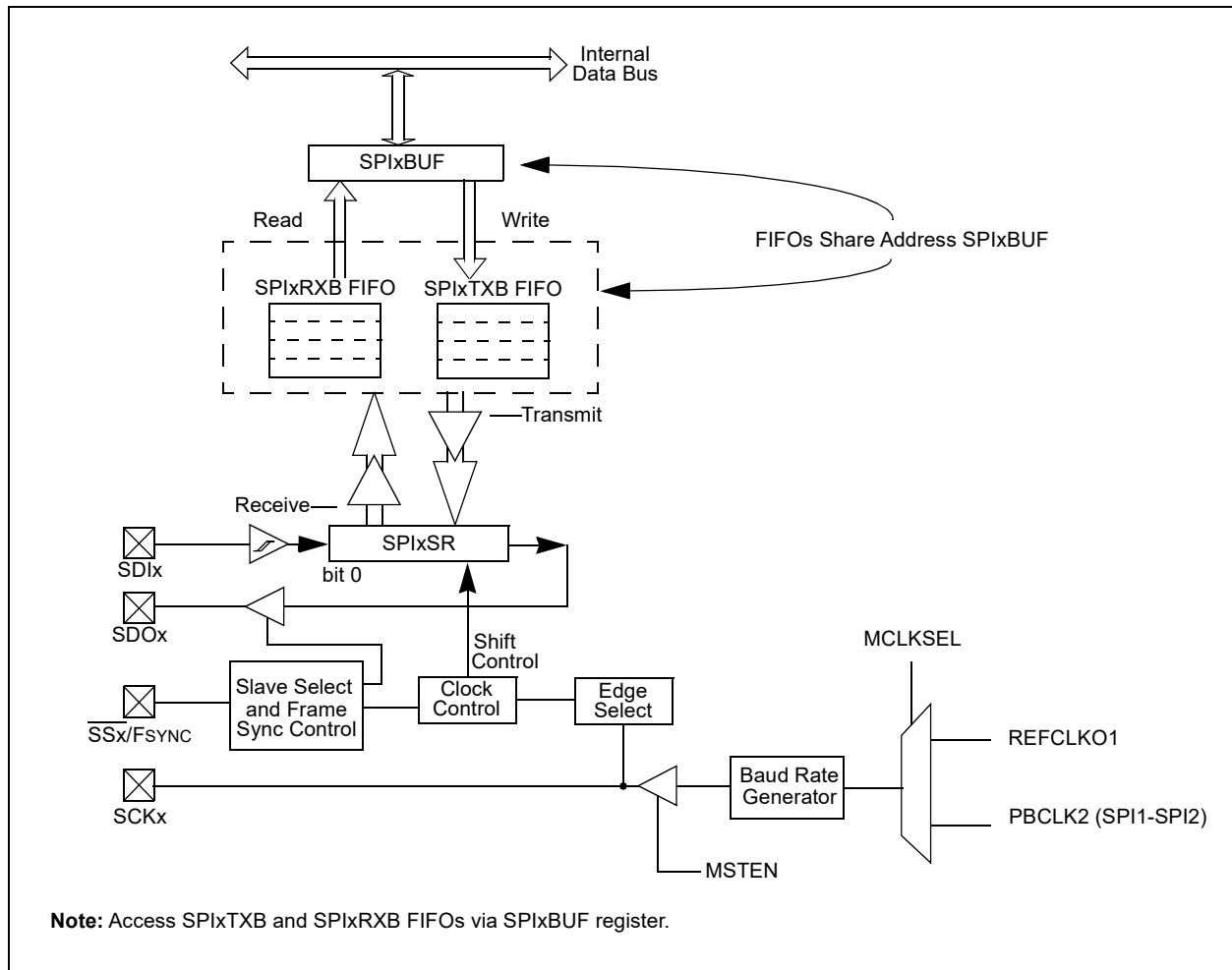
The SPI/I<sup>2</sup>S module is a synchronous serial interface that is useful for communicating with external peripherals and other microcontroller devices, as well as digital audio devices. These peripheral devices may be Serial EEPROMs, Shift registers, display drivers, analog-to-digital converters (ADC), and so on.

The SPI/I<sup>2</sup>S module is compatible with Motorola® SPI and SIOP interfaces.

The following are some of the key features of the SPI module:

- Master and Slave modes support
- Four different clock formats
- Enhanced Framed SPI protocol support
- User-configurable 32/24/16/8-bit data width
- Separate SPI FIFO buffers for receive and transmit
  - FIFO buffers act as 4/8/16-level deep FIFOs based on 32/24/16/8-bit data width
- Programmable interrupt event on every 8-bit, 16-bit and 32-bit data transfer
- Operation during Sleep and Idle modes
- Audio codec support:
  - I<sup>2</sup>S protocol
  - Left-justified
  - Right-justified
  - PCM

**FIGURE 17-1: SPI/I<sup>2</sup>S MODULE BLOCK DIAGRAM**



## 17.1 SPI Control Registers

**TABLE 17-1: SPI1 AND SPI2 REGISTER MAP**

| Virtual Address<br>(BF82_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits          |         |        |               |              |              |        |        |         |        |               |        |              |      |              | All Resets |      |
|-----------------------------|---------------------------------|-----------|---------------|---------|--------|---------------|--------------|--------------|--------|--------|---------|--------|---------------|--------|--------------|------|--------------|------------|------|
|                             |                                 |           | 31/15         | 30/14   | 29/13  | 28/12         | 27/11        | 26/10        | 25/9   | 24/8   | 23/7    | 22/6   | 21/5          | 20/4   | 19/3         | 18/2 | 17/1         |            | 16/0 |
| 7000                        | SPI1CON                         | 31:16     | FRMEN         | FRMSYNC | FRMPOL | MSEN          | FRMSYPW      | FRMCNT<2:0>  |        |        | MCLKSEL | —      | —             | —      | —            | —    | SPIFE        | ENHBUF     | 0000 |
|                             |                                 | 15:0      | ON            | —       | SIDL   | DISSDO        | MODE32       | MODE16       | SMP    | CKE    | SSEN    | CKP    | MSTEN         | DISSDI | STXISEL<1:0> |      | SRXISEL<1:0> |            | 0000 |
| 7010                        | SPI1STAT                        | 31:16     | —             | —       | —      | RXBUFELM<4:0> |              |              |        | —      | —       | —      | TXBUFELM<4:0> |        |              |      | 0000         |            |      |
|                             |                                 | 15:0      | —             | —       | —      | FRMERR        | SPIBUSY      | —            | —      | SPITUR | SRMT    | SPIROV | SPIRBE        | —      | SPITBE       | —    | SPITBF       | SPIRBF     | 0028 |
| 7020                        | SPI1BUF                         | 31:16     | DATA<31:0>    |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
|                             |                                 | 15:0      |               |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
| 7030                        | SPI1BRG                         | 31:16     | —             | —       | —      | —             | —            | —            | —      | —      | —       | —      | —             | —      | —            | —    | —            | —          | 0000 |
|                             |                                 | 15:0      | BRG<12:0>     |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
| 7040                        | SPI1CON2                        | 31:16     | —             | —       | —      | —             | —            | —            | —      | —      | —       | —      | —             | —      | —            | —    | —            | —          | 0000 |
|                             |                                 | 15:0      | SPI<br>SGNEXT | —       | —      | FRM<br>ERREN  | SPI<br>ROVEN | SPI<br>TUREN | IGNROV | IGNTUR | AUDEN   | —      | —             | —      | AUD<br>MONO  | —    | AUDMOD<1:0>  |            | 0C00 |
| 7200                        | SPI2CON                         | 31:16     | FRMEN         | FRMSYNC | FRMPOL | MSEN          | FRMSYPW      | FRMCNT<2:0>  |        |        | MCLKSEL | —      | —             | —      | —            | —    | SPIFE        | ENHBUF     | 0000 |
|                             |                                 | 15:0      | ON            | —       | SIDL   | DISSDO        | MODE32       | MODE16       | SMP    | CKE    | SSEN    | CKP    | MSTEN         | DISSDI | STXISEL<1:0> |      | SRXISEL<1:0> |            | 0000 |
| 7210                        | SPI2STAT                        | 31:16     | —             | —       | —      | RXBUFELM<4:0> |              |              |        | —      | —       | —      | TXBUFELM<4:0> |        |              |      | 0000         |            |      |
|                             |                                 | 15:0      | —             | —       | —      | FRMERR        | SPIBUSY      | —            | —      | SPITUR | SRMT    | SPIROV | SPIRBE        | —      | SPITBE       | —    | SPITBF       | SPIRBF     | 0028 |
| 7220                        | SPI2BUF                         | 31:16     | DATA<31:0>    |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
|                             |                                 | 15:0      |               |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
| 7230                        | SPI2BRG                         | 31:16     | —             | —       | —      | —             | —            | —            | —      | —      | —       | —      | —             | —      | —            | —    | —            | —          | 0000 |
|                             |                                 | 15:0      | BRG<12:0>     |         |        |               |              |              |        |        |         |        |               |        |              |      |              | 0000       |      |
| 7240                        | SPI2CON2                        | 31:16     | —             | —       | —      | —             | —            | —            | —      | —      | —       | —      | —             | —      | —            | —    | —            | —          | 0000 |
|                             |                                 | 15:0      | SPI<br>SGNEXT | —       | —      | FRM<br>ERREN  | SPI<br>ROVEN | SPI<br>TUREN | IGNROV | IGNTUR | AUDEN   | —      | —             | —      | AUD<br>MONO  | —    | AUDMOD<1:0>  |            | 0C00 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table except SPIxBUF have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 17-1: SPIxCON: SPI CONTROL REGISTER (X=1-2)**

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6     | Bit 29/21/13/5 | Bit 28/20/12/4        | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0         |
|-----------|------------------------|--------------------|----------------|-----------------------|----------------|----------------|---------------|-----------------------|
| 31:24     | R/W-0                  | R/W-0              | R/W-0          | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0                 |
|           | FRMEN                  | FRMSYNC            | FRMPOL         | MSEN                  | FRMSYPW        | FRMCNT<2:0>    |               |                       |
| 23:16     | R/W-0                  | U-0                | U-0            | U-0                   | U-0            | U-0            | R/W-0         | R/W-0                 |
|           | MCLKSEL <sup>(1)</sup> | —                  | —              | —                     | —              | —              | SPIFE         | ENHBUF <sup>(1)</sup> |
| 15:8      | R/W-0                  | U-0                | R/W-0          | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0                 |
|           | ON                     | —                  | SIDL           | DISSDO <sup>(4)</sup> | MODE32         | MODE16         | SMP           | CKE <sup>(2)</sup>    |
| 7:0       | R/W-0                  | R/W-0              | R/W-0          | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0                 |
|           | SSEN                   | CKP <sup>(3)</sup> | MSTEN          | DISSDI <sup>(4)</sup> | STXISEL<1:0>   |                | SRXISEL<1:0>  |                       |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31     **FRMEN:** Framed SPI Support bit  
1 = Framed SPI support is enabled ( $\overline{SSx}$  pin used as FSYNC input/output)  
0 = Framed SPI support is disabled
- bit 30     **FRMSYNC:** Frame Sync Pulse Direction Control on  $\overline{SSx}$  pin bit (Framed SPI mode only)  
1 = Frame sync pulse input (Slave mode)  
0 = Frame sync pulse output (Master mode)
- bit 29     **FRMPOL:** Frame Sync Polarity bit (Framed SPI mode only)  
1 = Frame pulse is active-high  
0 = Frame pulse is active-low
- bit 28     **MSEN:** Master Mode Slave Select Enable bit  
1 = Slave select SPI support enabled. The  $\overline{SS}$  pin is automatically driven during transmission in Master mode. Polarity is determined by the FRMPOL bit.  
0 = Slave select SPI support is disabled.
- bit 27     **FRMSYPW:** Frame Sync Pulse Width bit  
1 = Frame sync pulse is one character wide  
0 = Frame sync pulse is one clock wide
- bit 26-24 **FRMCNT<2:0>:** Frame Sync Pulse Counter bits. Controls the number of data characters transmitted per pulse. This bit is only valid in Framed mode.  
111 = Reserved  
110 = Reserved  
101 = Generate a frame sync pulse on every 32 data characters  
100 = Generate a frame sync pulse on every 16 data characters  
011 = Generate a frame sync pulse on every 8 data characters  
010 = Generate a frame sync pulse on every 4 data characters  
001 = Generate a frame sync pulse on every 2 data characters  
000 = Generate a frame sync pulse on every data character
- bit 23     **MCLKSEL:** Master Clock Enable bit<sup>(1)</sup>  
1 = REFCLKO1 is used by the Baud Rate Generator  
0 = PBCLK2 is used by the Baud Rate Generator for SPI1 and SPI2
- bit 22-18 **Unimplemented:** Read as '0'

- Note 1:** This bit can only be written when the ON bit = 0. Refer to **36.0 “Electrical Characteristics”** for maximum clock frequency requirements.
- 2:** This bit is not used in the Framed SPI mode. The user should program this bit to '0' for the Framed SPI mode (FRMEN = 1).
- 3:** When AUDEN = 1, the SPI/I<sup>2</sup>S module functions as if the CKP bit is equal to '1', regardless of the actual value of the CKP bit.
- 4:** This bit present for legacy compatibility and is superseded by PPS functionality on these devices (see **10.3 “Peripheral Pin Select (PPS)”** for more information).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 17-1: SPIxCON: SPI CONTROL REGISTER (CONTINUED)(X=1-2)

- bit 17 **SPIFE:** Frame Sync Pulse Edge Select bit (Framed SPI mode only)  
1 = Frame synchronization pulse coincides with the first bit clock  
0 = Frame synchronization pulse precedes the first bit clock
- bit 16 **ENHBUF:** Enhanced Buffer Enable bit<sup>(1)</sup>  
1 = Enhanced Buffer mode is enabled  
0 = Enhanced Buffer mode is disabled
- bit 15 **ON:** SPI/I<sup>2</sup>S Module On bit  
1 = SPI/I<sup>2</sup>S module is enabled  
0 = SPI/I<sup>2</sup>S module is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **SIDL:** Stop in Idle Mode bit  
1 = Discontinue operation when CPU enters in Idle mode  
0 = Continue operation in Idle mode
- bit 12 **DISSDO:** Disable SDOx pin bit<sup>(4)</sup>  
1 = SDOx pin is not used by the module. Pin is controlled by associated PORT register  
0 = SDOx pin is controlled by the module
- bit 11-10 **MODE<32,16>:** 32/16-Bit Communication Select bits  
When AUDEN = 1:  
MODE32 MODE16 Communication  
11 24-bit Data, 32-bit FIFO, 32-bit Channel/64-bit Frame  
10 32-bit Data, 32-bit FIFO, 32-bit Channel/64-bit Frame  
01 16-bit Data, 16-bit FIFO, 32-bit Channel/64-bit Frame  
00 16-bit Data, 16-bit FIFO, 16-bit Channel/32-bit Frame
- When AUDEN = 0:  
MODE32 MODE16 Communication  
1x 32-bit  
01 16-bit  
00 8-bit
- bit 9 **SMP:** SPI Data Input Sample Phase bit  
Master mode (MSTEN = 1):  
1 = Input data sampled at end of data output time  
0 = Input data sampled at middle of data output time  
Slave mode (MSTEN = 0):  
SMP value is ignored when SPI is used in Slave mode. The module always uses SMP = 0.
- bit 8 **CKE:** SPI Clock Edge Select bit<sup>(2)</sup>  
1 = Serial output data changes on transition from active clock state to Idle clock state (see CKP bit)  
0 = Serial output data changes on transition from Idle clock state to active clock state (see CKP bit)
- bit 7 **SSEN:** Slave Select Enable (Slave mode) bit  
1 = SSx pin used for Slave mode  
0 = SSx pin not used for Slave mode, pin controlled by port function.
- bit 6 **CKP:** Clock Polarity Select bit<sup>(3)</sup>  
1 = Idle state for clock is a high level; active state is a low level  
0 = Idle state for clock is a low level; active state is a high level

- Note 1:** This bit can only be written when the ON bit = 0. Refer to [36.0 “Electrical Characteristics”](#) for maximum clock frequency requirements.
- 2:** This bit is not used in the Framed SPI mode. The user should program this bit to '0' for the Framed SPI mode (FRMEN = 1).
- 3:** When AUDEN = 1, the SPI/I<sup>2</sup>S module functions as if the CKP bit is equal to '1', regardless of the actual value of the CKP bit.
- 4:** This bit present for legacy compatibility and is superseded by PPS functionality on these devices (see [10.3 “Peripheral Pin Select \(PPS\)”](#) for more information).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 17-1: SPIxCON: SPI CONTROL REGISTER (CONTINUED)(X=1-2)

- bit 5     **MSTEN**: Master Mode Enable bit  
          1 = Master mode  
          0 = Slave mode
- bit 4     **DISSDI**: Disable SDI bit<sup>(4)</sup>  
          1 = SDI pin is not used by the SPI module (pin is controlled by PORT function)  
          0 = SDI pin is controlled by the SPI module
- bit 3-2   **STXISEL<1:0>**: SPI Transmit Buffer Empty Interrupt Mode bits  
          11 = Interrupt is generated when the buffer is not full (has one or more empty elements)  
          10 = Interrupt is generated when the buffer is empty by one-half or more  
          01 = Interrupt is generated when the buffer is completely empty  
          00 = Interrupt is generated when the last transfer is shifted out of SPISR and transmit operations are complete
- bit 1-0   **SRXISEL<1:0>**: SPI Receive Buffer Full Interrupt Mode bits  
          11 = Interrupt is generated when the buffer is full  
          10 = Interrupt is generated when the buffer is full by one-half or more  
          01 = Interrupt is generated when the buffer is not empty  
          00 = Interrupt is generated when the last word in the receive buffer is read (i.e., buffer is empty)

- Note 1:** This bit can only be written when the ON bit = 0. Refer to [36.0 “Electrical Characteristics”](#) for maximum clock frequency requirements.
- 2:** This bit is not used in the Framed SPI mode. The user should program this bit to ‘0’ for the Framed SPI mode (FRMEN = 1).
- 3:** When AUDEN = 1, the SPI/I<sup>2</sup>S module functions as if the CKP bit is equal to ‘1’, regardless of the actual value of the CKP bit.
- 4:** This bit present for legacy compatibility and is superseded by PPS functionality on these devices (see [10.3 “Peripheral Pin Select \(PPS\)”](#) for more information).



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 17-3: SPIxSTAT: SPI STATUS REGISTER (X=1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | —              | —              | —              | RXBUFELM<4:0>  |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | —              | —              | —              | TXBUFELM<4:0>  |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/C-0, HS      | R-0, HS, HC    | U-0            | U-0           | R-0           |
|           | —              | —              | —              | FRMERR         | SPIBUSY        | —              | —             | SPITUR        |
| 7:0       | R-0, HS, HC    | R/C-0, HS      | R-1, HS, HC    | U-0            | R-1, HS, HC    | U-0            | R-0, HS, HC   | R-0, HS, HC   |
|           | SRMT           | SPIROV         | SPIRBE         | —              | SPITBE         | —              | SPITBF        | SPIRBF        |

|                   |                          |                                    |                    |
|-------------------|--------------------------|------------------------------------|--------------------|
| <b>Legend:</b>    | HC = Cleared in hardware | HS = Set in hardware               | C = Clearable bit  |
| R = Readable bit  | W = Writable bit         | U = Unimplemented bit, read as '0' |                    |
| -n = Value at POR | '1' = Bit is set         | '0' = Bit is cleared               | x = Bit is unknown |

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **RXBUFELM<4:0>**: Receive Buffer Element Count bits (valid only when ENHBUF = 1)

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TXBUFELM<4:0>**: Transmit Buffer Element Count bits (valid only when ENHBUF = 1)

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **FRMERR**: SPI Frame Error status bit

1 = Frame error is detected

0 = No Frame error is detected

This bit is only valid when FRMEN = 1.

bit 11 **SPIBUSY**: SPI Activity Status bit

1 = SPI peripheral is currently busy with some transactions

0 = SPI peripheral is currently idle

bit 10-9 **Unimplemented:** Read as '0'

bit 8 **SPITUR**: Transmit Under Run bit

1 = Transmit buffer has encountered an underrun condition

0 = Transmit buffer has no underrun condition

This bit is only valid in Framed Sync mode; the underrun condition must be cleared by disabling/re-enabling the module.

bit 7 **SRMT**: Shift Register Empty bit (valid only when ENHBUF = 1)

1 = When SPI module shift register is empty

0 = When SPI module shift register is not empty

bit 6 **SPIROV**: Receive Overflow Flag bit

1 = A new data is completely received and discarded. The user software has not read the previous data in the SPIxBUF register.

0 = No overflow has occurred

This bit is set in hardware; can only be cleared (= 0) in software.

bit 5 **SPIRBE**: RX FIFO Empty bit (valid only when ENHBUF = 1)

1 = RX FIFO is empty (CRPTR = SWPTR)

0 = RX FIFO is not empty (CRPTR ≠ SWPTR)

bit 4 **Unimplemented:** Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 17-3: SPIxSTAT: SPI STATUS REGISTER (X=1-2)

- bit 3     **SPITBE:** SPI Transmit Buffer Empty Status bit  
          1 = Transmit buffer, SPIxTXB is empty  
          0 = Transmit buffer, SPIxTXB is not empty  
          Automatically set in hardware when SPI transfers data from SPIxTXB to SPIxSR.  
          Automatically cleared in hardware when SPIxBUF is written to, loading SPIxTXB.
- bit 2     **Unimplemented:** Read as '0'
- bit 1     **SPITBF:** SPI Transmit Buffer Full Status bit  
          1 = Transmit not yet started, SPITXB is full  
          0 = Transmit buffer is not full  
          Standard Buffer Mode:  
          Automatically set in hardware when the core writes to the SPIBUF location, loading SPITXB.  
          Automatically cleared in hardware when the SPI module transfers data from SPITXB to SPISR.  
          Enhanced Buffer Mode:  
          Set when CWPTR + 1 = SRPTR; cleared otherwise
- bit 0     **SPIRBF:** SPI Receive Buffer Full Status bit  
          1 = Receive buffer, SPIxRXB is full  
          0 = Receive buffer, SPIxRXB is not full  
          Standard Buffer Mode:  
          Automatically set in hardware when the SPI module transfers data from SPIxSR to SPIxRXB.  
          Automatically cleared in hardware when SPIxBUF is read from, reading SPIxRXB.  
          Enhanced Buffer Mode:  
          Set when SWPTR + 1 = CRPTR; cleared otherwise



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 17-4: SPIxBUF: SPIx BUFFER REGISTER (x = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DATA<31:24>    |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DATA<23:16>    |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DATA<15:8>     |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DATA<7:0>      |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0      **DATA<31:0>** FIFO Data bits

When MODE32 or MODE16 selects 32-bit data, the SPI uses DATA<31:0>.

When MODE32 or MODE16 selects 24-bit data, the SPI only uses DATA<24:0>.

When MODE32 or MODE16 selects 16-bit data, the SPI only uses DATA<15:0>.

When MODE32 or MODE16 selects 8-bit data, the SPI only uses DATA<7:0>.

**REGISTER 17-5: SPIxBRG: SPIx BAUD RATE GENERATOR REGISTER (x= 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | BRG<12:8>      |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | BRG<7:0>       |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-13      **Unimplemented:** Read as '0'

bit 12-0      **BRG<12:0>** Baud Rate Generator Divisor bits

Baud Rate =  $FPBCLKx / (2 * (SPIxBRG + 1))$ , where 'x' = 1, 2. Therefore, the maximum baud rate possible is  $FPBCLKx / 2$  (SPIxBRG = 0) and the minimum baud rate possible is  $FPBCLKx / 16384$ .

**Note:** Changing the BRG value when the ON bit is equal to '1' causes undefined behavior.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

---

## 18.0 INTER-INTEGRATED CIRCUIT (I<sup>2</sup>C)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 24. “Inter-Integrated Circuit (I<sup>2</sup>C)”** (DS60001116), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The I<sup>2</sup>C module provides complete hardware support for both Slave and Multi-Master modes of the I<sup>2</sup>C serial communication standard.

Each I<sup>2</sup>C module has a 2-pin interface:

- SCLx pin is clock
- SDAx pin is data

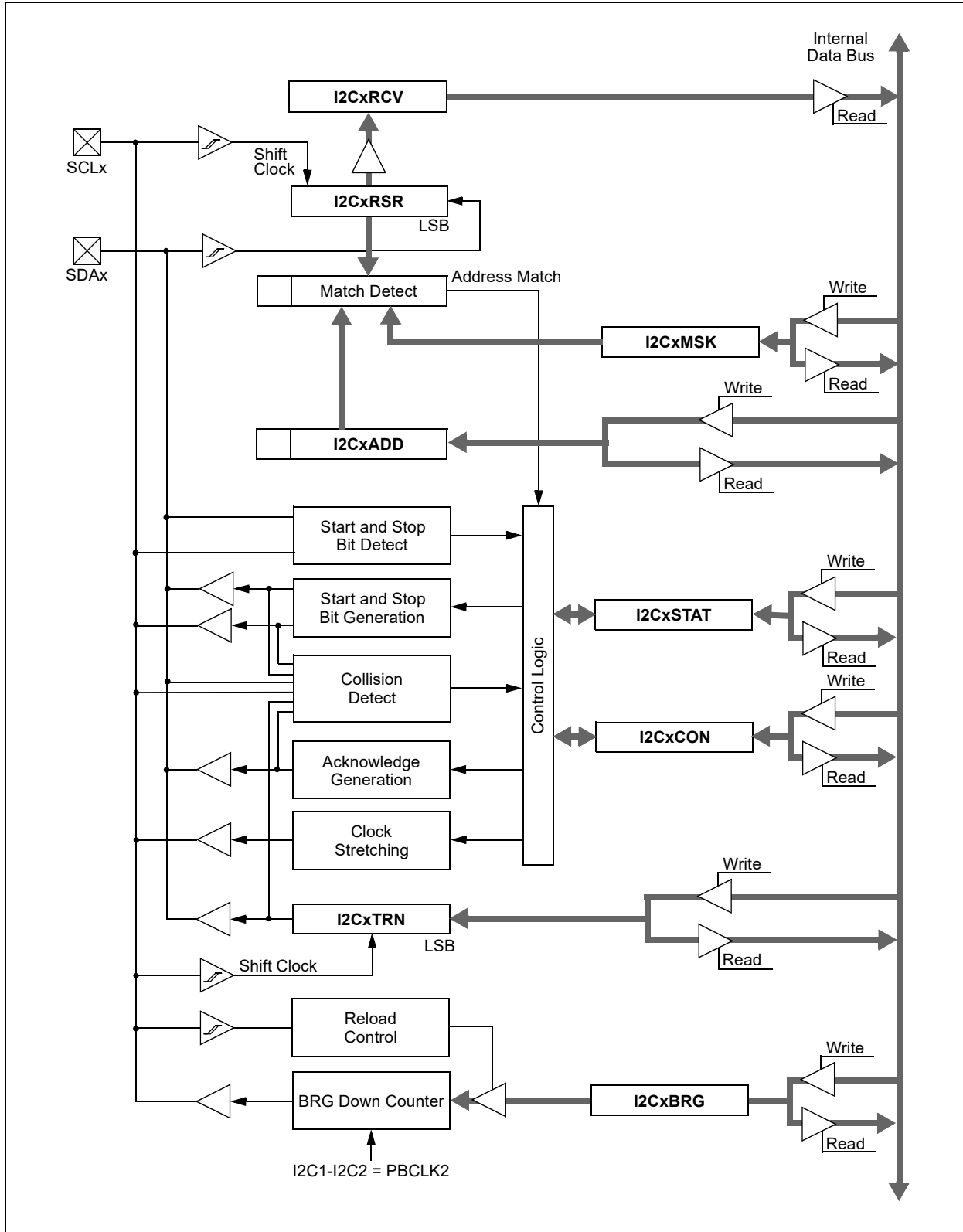
Each I<sup>2</sup>C module offers the following key features:

- I<sup>2</sup>C interface supporting both master and slave operation
- I<sup>2</sup>C Slave mode supports 7-bit and 10-bit addressing
- I<sup>2</sup>C Master mode supports 7-bit and 10-bit addressing
- I<sup>2</sup>C port allows bidirectional transfers between master and slaves
- Serial clock synchronization for the I<sup>2</sup>C port can be used as a handshake mechanism to suspend and resume serial transfer (SCLREL control)
- I<sup>2</sup>C supports multi-master operation; detects bus collision and arbitrates accordingly
- Provides support for address bit masking
- SMBus support

[Figure 18-1](#) illustrates the I<sup>2</sup>C module block diagram.

# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 18-1: I<sup>2</sup>C BLOCK DIAGRAM



## 18.1 I<sup>2</sup>C Control Registers

**TABLE 18-1: I2C1 AND I2C2 REGISTER MAP**

| Virtual Address<br>BF82_# | Register<br>Name(1) | Bit Range | Bits    |        |        |        |        |       |        |       |       |       |       |       |      |       |       |      | All Resets |      |
|---------------------------|---------------------|-----------|---------|--------|--------|--------|--------|-------|--------|-------|-------|-------|-------|-------|------|-------|-------|------|------------|------|
|                           |                     |           | 31/15   | 30/14  | 29/13  | 28/12  | 27/11  | 26/10 | 25/9   | 24/8  | 23/7  | 22/6  | 21/5  | 20/4  | 19/3 | 18/2  | 17/1  | 16/0 |            |      |
| 6000                      | I2C1CON             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | PCIE  | SCIE  | BOEN | SDAHT | SBCDE | AHEN | DHEN       | 0000 |
|                           |                     | 15:0      | ON      | —      | SIDL   | SCLREL | STRICT | —     | DISSLW | SMEN  | GCEN  | STREN | ACKDT | ACKEN | RCEN | PEN   | RSEN  | SEN  | 1000       |      |
| 6010                      | I2C1STAT            | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | ACKSTAT | TRSTAT | ACKTIM | —      | —      | BCL   | GCSTAT | ADD10 | IWCOL | I2COV | D/A   | P     | S    | R/W   | RBF   | TBF  | 0000       |      |
| 6020                      | I2C1ADD             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6030                      | I2C1MSK             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6040                      | I2C1BRG             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6050                      | I2C1TRN             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6060                      | I2C1RCV             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6200                      | I2C2CON             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | PCIE  | SCIE  | BOEN | SDAHT | SBCDE | AHEN | DHEN       | 0000 |
|                           |                     | 15:0      | ON      | —      | SIDL   | SCLREL | STRICT | —     | DISSLW | SMEN  | GCEN  | STREN | ACKDT | ACKEN | RCEN | PEN   | RSEN  | SEN  | 1000       |      |
| 6210                      | I2C2STAT            | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | ACKSTAT | TRSTAT | ACKTIM | —      | —      | BCL   | GCSTAT | ADD10 | IWCOL | I2COV | D/A   | P     | S    | R/W   | RBF   | TBF  | 0000       |      |
| 6220                      | I2C2ADD             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6230                      | I2C2MSK             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6240                      | I2C2BRG             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6250                      | I2C2TRN             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
| 6260                      | I2C2RCV             | 31:16     | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |
|                           |                     | 15:0      | —       | —      | —      | —      | —      | —     | —      | —     | —     | —     | —     | —     | —    | —     | —     | —    | —          | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table except I2CxRCV have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 18-1: I2CxCON: I<sup>2</sup>C CONTROL REGISTER (X=1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | PCIE           | SCIE           | BOEN           | SDAHT          | SBCDE          | AHEN          | DHEN          |
| 15:8      | R/W-0          | U-0            | R/W-0          | R/W-1, HC      | R/W-0          | r-0            | R/W-0         | R/W-0         |
|           | ON             | —              | SIDL           | SCKREL         | STRICT         | —              | DISSLW        | SMEN          |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0, HC      | R/W-0, HC      | R/W-0, HC      | R/W-0, HC     | R/W-0, HC     |
|           | GCEN           | STREN          | ACKDT          | ACKEN          | RCEN           | PEN            | RSEN          | SEN           |

|                   |                          |                                    |
|-------------------|--------------------------|------------------------------------|
| <b>Legend:</b>    | HC = Cleared in Hardware | r = Reserved bit                   |
| R = Readable bit  | W = Writable bit         | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set         | '0' = Bit is cleared               |
|                   |                          | x = Bit is unknown                 |

bit 31-23 **Unimplemented:** Read as '0'

bit 22 **PCIE:** Stop Condition Interrupt Enable bit (I<sup>2</sup>C Slave mode only)

1 = Enable interrupt on detection of Stop condition

0 = Stop detection interrupts are disabled

bit 21 **SCIE:** Start Condition Interrupt Enable bit (I<sup>2</sup>C Slave mode only)

1 = Enable interrupt on detection of Start or Restart conditions

0 = Start detection interrupts are disabled

bit 20 **BOEN:** Buffer Overwrite Enable bit (I<sup>2</sup>C Slave mode only)

1 = I2CxRCV is updated and  $\overline{ACK}$  is generated for a received address/data byte, ignoring the state of the I2COV bit (I2CxSTAT<6>) only if the RBF bit (I2CxSTAT<2>) = 0

0 = I2CxRCV is only updated when the I2COV bit (I2CxSTAT<6>) is clear

bit 19 **SDAHT:** SDA Hold Time Selection bit

1 = Minimum of 300 ns hold time on SDA after the falling edge of SCL

0 = Minimum of 100 ns hold time on SDA after the falling edge of SCL

bit 18 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I<sup>2</sup>C Slave mode only)

1 = Enable slave bus collision interrupts

0 = Slave bus collision interrupts are disabled

bit 17 **AHEN:** Address Hold Enable bit (Slave mode only)

1 = Following the 8th falling edge of SCL for a matching received address byte; SCKREL bit will be cleared and the SCL will be held low.

0 = Address holding is disabled

bit 16 **DHEN:** Data Hold Enable bit (I<sup>2</sup>C Slave mode only)

1 = Following the 8th falling edge of SCL for a received data byte; slave hardware clears the SCKREL bit and SCL is held low

0 = Data holding is disabled

bit 15 **ON:** I<sup>2</sup>C Enable bit

1 = Enables the I<sup>2</sup>C module and configures the SDA and SCL pins as serial port pins

0 = Disables the I<sup>2</sup>C module; all I<sup>2</sup>C pins are controlled by PORT functions

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 18-1: I2CxCON: I<sup>2</sup>C CONTROL REGISTER (CONTINUED) (X=1-2)

- bit 12 **SCLREL:** SCLx Release Control bit (when operating as I<sup>2</sup>C slave)  
1 = Release SCLx clock  
0 = Hold SCLx clock low (clock stretch)  
If STREN = 1:  
Bit is R/W (i.e., software can write '0' to initiate stretch and write '1' to release clock). Hardware clear at beginning of slave transmission. Hardware clear at end of slave reception.  
If STREN = 0:  
Bit is R/S (i.e., software can only write '1' to release clock). Hardware clear at beginning of slave transmission.
- bit 11 **STRICT:** Strict I<sup>2</sup>C Reserved Address Rule Enable bit  
1 = Strict reserved addressing is enforced. Device does not respond to reserved address space or generate addresses in reserved address space.  
0 = Strict I<sup>2</sup>C Reserved Address Rule not enabled
- bit 10 **Reserved**
- bit 9 **DISSLW:** Disable Slew Rate Control bit  
1 = Slew rate control is disabled  
0 = Slew rate control is enabled
- bit 8 **SMEN:** SMBus Input Levels bit  
1 = Enable I/O pin thresholds compliant with SMBus specification  
0 = Disable SMBus input thresholds
- bit 7 **GCEN:** General Call Enable bit (when operating as I<sup>2</sup>C slave)  
1 = Enable interrupt when a general call address is received in the I2CxRSR (module is enabled for reception)  
0 = General call address disabled
- bit 6 **STREN:** SCLx Clock Stretch Enable bit (when operating as I<sup>2</sup>C slave)  
Used in conjunction with SCLREL bit.  
1 = Enable software or receive clock stretching  
0 = Disable software or receive clock stretching
- bit 5 **ACKDT:** Acknowledge Data bit (when operating as I<sup>2</sup>C master, applicable during master receive)  
Value that is transmitted when the software initiates an Acknowledge sequence.  
1 = Send NACK during Acknowledge  
0 = Send ACK during Acknowledge
- bit 4 **ACKEN:** Acknowledge Sequence Enable bit (when operating as I<sup>2</sup>C master, applicable during master receive)  
1 = Initiate Acknowledge sequence on SDAx and SCLx pins and transmit ACKDT data bit. Hardware clear at end of master Acknowledge sequence.  
0 = Acknowledge sequence not in progress
- bit 3 **RCEN:** Receive Enable bit (when operating as I<sup>2</sup>C master)  
1 = Enables Receive mode for I<sup>2</sup>C. Hardware clear at end of eighth bit of master receive data byte.  
0 = Receive sequence not in progress
- bit 2 **PEN:** Stop Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Stop condition on SDAx and SCLx pins. Hardware clear at end of master Stop sequence.  
0 = Stop condition not in progress
- bit 1 **RSEN:** Repeated Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Repeated Start condition on SDAx and SCLx pins. Hardware clear at end of master Repeated Start sequence.  
0 = Repeated Start condition not in progress
- bit 0 **SEN:** Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Start condition on SDAx and SCLx pins. Hardware clear at end of master Start sequence.  
0 = Start condition not in progress

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 18-2: I2CxSTAT: I<sup>2</sup>C STATUS REGISTER (X=1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R-0, HS, HC    | R-0, HS, HC    | R/C-0, HS, HC  | U-0            | U-0            | R/C-0, HS      | R-0, HS, HC   | R-0, HS, HC   |
|           | ACKSTAT        | TRSTAT         | ACKTIM         | —              | —              | BCL            | GCSTAT        | ADD10         |
| 7:0       | R/C-0, HS, SC  | R/C-0, HS, SC  | R-0, HS, HC    | R/C-0, HS, HC  | R/C-0, HS, HC  | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | IWCOL          | I2COV          | D_A            | P              | S              | R_W            | RBF           | TBF           |

|                   |                   |                                    |                       |
|-------------------|-------------------|------------------------------------|-----------------------|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared              | SC = Software Cleared |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0' |                       |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared               | C = Clearable bit     |

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ACKSTAT:** Acknowledge Status bit  
(when operating as I<sup>2</sup>C master, applicable to master transmit operation)

- 1 = NACK received from slave
- 0 = ACK received from slave

Hardware set or clear at end of slave Acknowledge.

bit 14 **TRSTAT:** Transmit Status bit (when operating as I<sup>2</sup>C master, applicable to master transmit operation)

- 1 = Master transmit is in progress (8 bits + ACK)
- 0 = Master transmit is not in progress

Hardware set at beginning of master transmission. Hardware clear at end of slave Acknowledge.

bit 13 **ACKTIM:** Acknowledge Time Status bit (Valid in I<sup>2</sup>C Slave mode only)

- 1 = I<sup>2</sup>C bus is in an Acknowledge sequence, set on 8th falling edge of SCL clock
- 0 = Not an Acknowledge sequence, cleared on 9th rising edge of SCL clock

bit 12-11 **Unimplemented:** Read as '0'

bit 10 **BCL:** Master Bus Collision Detect bit

- 1 = A bus collision has been detected during a master operation
- 0 = No collision

Hardware set at detection of bus collision.

bit 9 **GCSTAT:** General Call Status bit

- 1 = General call address was received
- 0 = General call address was not received

Hardware set when address matches general call address. Hardware clear at Stop detection.

bit 8 **ADD10:** 10-bit Address Status bit

- 1 = 10-bit address was matched
- 0 = 10-bit address was not matched

Hardware set at match of 2nd byte of matched 10-bit address. Hardware clear at Stop detection.

bit 7 **IWCOL:** Write Collision Detect bit

- 1 = An attempt to write the I2CxTRN register failed because the I<sup>2</sup>C module is busy
- 0 = No collision

Hardware set at occurrence of write to I2CxTRN while busy (cleared by software).

bit 6 **I2COV:** Receive Overflow Flag bit

- 1 = A byte was received while the I2CxRCV register is still holding the previous byte
- 0 = No overflow

Hardware set at attempt to transfer I2CxRSR to I2CxRCV (cleared by software).



# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 18-2: I2CxSTAT: I<sup>2</sup>C STATUS REGISTER (CONTINUED)(X=1-2)

- bit 5     **D\_A**: Data/Address bit (when operating as I<sup>2</sup>C slave)  
1 = Indicates that the last byte received was data  
0 = Indicates that the last byte received was device address  
Hardware clear at device address match. Hardware set by reception of slave byte.
- bit 4     **P**: Stop bit  
1 = Indicates that a Stop bit has been detected last  
0 = Stop bit was not detected last  
Hardware set or clear when Start, Repeated Start or Stop detected.
- bit 3     **S**: Start bit  
1 = Indicates that a Start (or Repeated Start) bit has been detected last  
0 = Start bit was not detected last  
Hardware set or clear when Start, Repeated Start or Stop detected.
- bit 2     **R\_W**: Read/Write Information bit (when operating as I<sup>2</sup>C slave)  
1 = Read – indicates data transfer is output from slave  
0 = Write – indicates data transfer is input to slave  
Hardware set or clear after reception of I<sup>2</sup>C device address byte.
- bit 1     **RBF**: Receive Buffer Full Status bit  
1 = Receive complete, I2CxRCV is full  
0 = Receive not complete, I2CxRCV is empty  
Hardware set when I2CxRCV is written with received byte. Hardware clear when software reads I2CxRCV.
- bit 0     **TBF**: Transmit Buffer Full Status bit  
1 = Transmit in progress, I2CxTRN is full  
0 = Transmit complete, I2CxTRN is empty  
Hardware set when software writes I2CxTRN. Hardware clear at completion of data transmission.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 18-3: I<sup>2</sup>CxBRG: – I<sup>2</sup>C MODULE BAUDRATE REGISTER (x=1-2)**

| Bit Range | Bit 31/23/15/7                 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|--------------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0                            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0                          |                |                |                |                |                |               |               |
|           | I2CxBRG<15:8> <sup>(1,2)</sup> |                |                |                |                |                |               |               |
| 7:0       | R/W-0                          |                |                |                |                |                |               |               |
|           | I2CxBRG<7:0> <sup>(1,2)</sup>  |                |                |                |                |                |               |               |

|                   |                   |                                    |                       |
|-------------------|-------------------|------------------------------------|-----------------------|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared              | SC = Software Cleared |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0' |                       |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared               | C = Clearable bit     |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **I2CxBRG<15:0>**: I<sup>2</sup>C Baud Rate Generator Value bits<sup>(1,2)</sup>  
 These bits control the divider function of the Peripheral Clock.

**EQUATION 18-1: BAUD RATE GENERATOR RELOAD VALUE CALCULATION**

$$I2CxBRG<15:0> = \frac{F_{PBCLK}}{2 - F_{SCK}} - 1 - \frac{F_{PBCLK} - 130ns}{2}$$

| PBCLK   | F <sub>sck</sub><br>(Two Rollovers of I2CxBRG) | Calculated<br>I2CxBRG<15:0> |
|---------|--|-----------------------------|
| 120 MHz | 1000 kHz                                       | 0x0034                      |
| 120 MHz | 400 kHz  | 0x008E                      |
| 120 MHz | 100 kHz  | 0x0250                      |
| 60MHz   | 1000 kHz                                       | 0x001A                      |
| 60MHz   | 400 kHz  | 0x0047                      |
| 60MHz   | 100 kHz  | 0x0128                      |

**EQUATION 18-2: SCK FREQUENCY**

$$F_{SCK} = \frac{F_{PBCLK}}{(2 \cdot I2CxBRG<15:0>) + 2 + (F_{PBCLK} - 130ns)}$$

**Note1:** The value of these bits must not be changed when the I<sup>2</sup>C module is active. It is safe to change the value when ON = 0, when idle or waiting in Master mode.

**2:** I2CxBRG values of 0x0 through 0x3 are expressly prohibited. Do not program the I2CxBRG register to any of these values, as indeterminate results may occur.

# PIC32MK GPG/MCJ with CAN FD Family

## 19.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 21. “Universal Asynchronous Receiver Transmitter (UART)”** (DS60001107), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

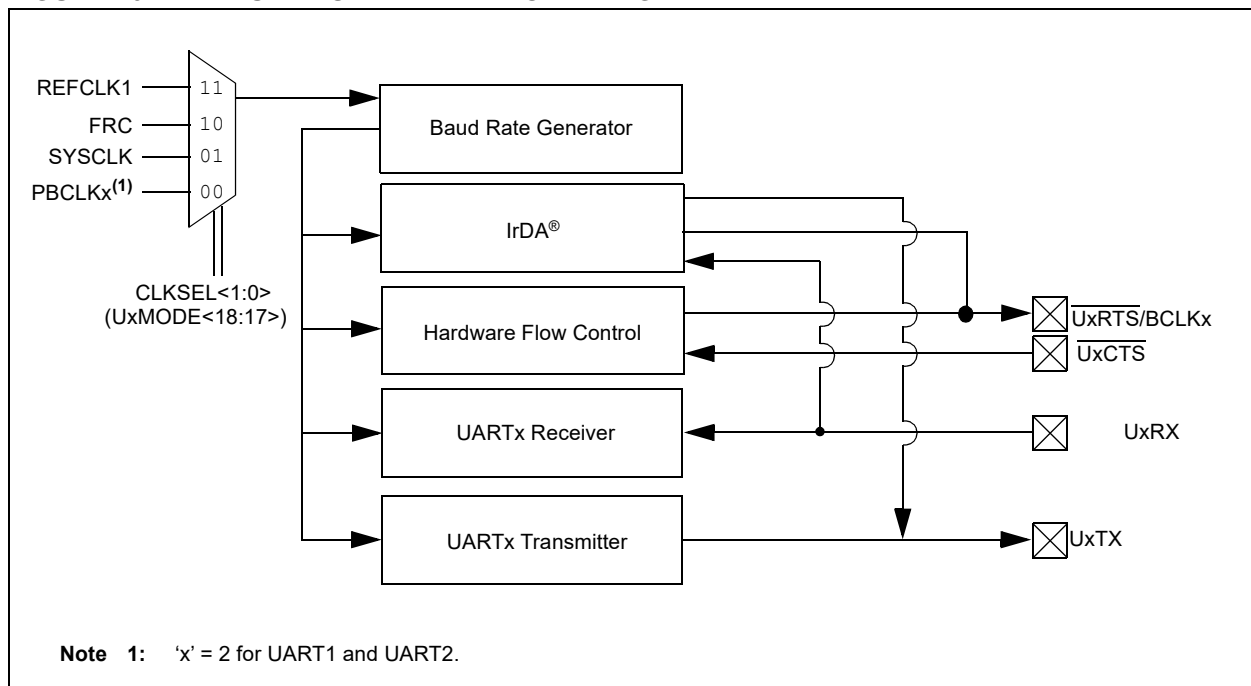
The UART module is one of the serial I/O modules available in PIC32MK GPG/MCJ with CAN FD Family of devices. The UART is a full-duplex, asynchronous communication channel that communicates with peripheral devices and personal computers through protocols, such as RS-232, RS-485, LIN, and IrDA®. The module also supports the hardware flow control option, with UxCTS and UxRTS pins, and also includes an IrDA encoder and decoder.

The following are key features of the UART module:

- Ability to receive data during Sleep mode
- Full-duplex, 8-bit or 9-bit data transmission
- Even, Odd, or No Parity options (for 8-bit data)
- One or two Stop bits
- Auto-baud support
- Four clock source inputs for asynchronous clocking
- Transmit and Receive (TX/RX) polarity control
- Hardware flow control option
- Fully integrated Baud Rate Generator (BRG) with 16-bit prescaler
- Baud rates up to 30 Mbps
- 8-level deep First-In-First-Out (FIFO) transmit data buffer
- 8-level deep FIFO receive data buffer
- Parity, framing and buffer overrun error detection
- Support for interrupt-only on address detect (ninth bit = 1)
- Separate transmit and receive interrupts
- Loopback mode for diagnostic support
- LIN Protocol support
- IrDA encoder and decoder with 16x baud clock output for external IrDA encoder/decoder support

Figure 19-1 illustrates a simplified block diagram of the UART module.

**FIGURE 19-1: UART SIMPLIFIED BLOCK DIAGRAM**



## 19.1 UART Control Registers

**TABLE 19-1: UART1 AND UART2 REGISTER MAP**

| Virtual Address<br>BF82_# | Register<br>Name      | Bit<br>Range | Bits         |        |       |        |       |       |      |                   |       |        |       |       |              |             | All<br>Resets |      |
|---------------------------|-----------------------|--------------|--------------|--------|-------|--------|-------|-------|------|-------------------|-------|--------|-------|-------|--------------|-------------|---------------|------|
|                           |                       |              | 31/15        | 30/14  | 29/13 | 28/12  | 27/11 | 26/10 | 25/9 | 24/8              | 23/7  | 22/6   | 21/5  | 20/4  | 19/3         | 18/2        |               | 17/1 |
| 8000                      | U1MODE <sup>(1)</sup> | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | SLPEN | CKRDY  | —     | —     | —            | CLKSEL<1:0> | RUNOV         | 0000 |
|                           |                       | 15:0         | ON           | —      | SIDL  | IREN   | RTSMD | —     | —    | UEN<1:0>          | WAKE  | LPBACK | ABAUD | RXINV | BRGH         | PDSEL<1:0>  | STSEL         | 0000 |
| 8010                      | U1STA <sup>(1)</sup>  | 31:16        | ADDRMSK<7:0> |        |       |        |       |       |      | ADDR<7:0>         |       |        |       |       |              |             | 0000          |      |
|                           |                       | 15:0         | UTXISEL<1:0> | UTXINV | URXEN | UTXBRK | UTXEN | UTXBF | TRMT | URXISEL<1:0>      | ADDEN | RIDLE  | PERR  | FERR  | OERR         | URXDA       | 0110          |      |
| 8020                      | U1TXREG               | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | —            | —           | —             | 0000 |
|                           |                       | 15:0         | —            | —      | —     | —      | —     | —     | TX8  | Transmit Register |       |        |       |       |              |             | 0000          |      |
| 8030                      | U1RXREG               | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | —            | —           | —             | 0000 |
|                           |                       | 15:0         | —            | —      | —     | —      | —     | —     | RX8  | Receive Register  |       |        |       |       |              |             | 0000          |      |
| 8040                      | U1BRG <sup>(1)</sup>  | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | U1BRG<19:16> |             | 0000          |      |
|                           |                       | 15:0         | U1BRG<15:0>  |        |       |        |       |       |      |                   |       |        |       |       |              |             | 0000          |      |
| 8200                      | U2MODE <sup>(1)</sup> | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | SLPEN | CKRDY  | —     | —     | —            | CLKSEL<1:0> | RUNOV         | 0000 |
|                           |                       | 15:0         | ON           | —      | SIDL  | IREN   | RTSMD | —     | —    | UEN<1:0>          | WAKE  | LPBACK | ABAUD | RXINV | BRGH         | PDSEL<1:0>  | STSEL         | 0000 |
| 8210                      | U2STA <sup>(1)</sup>  | 31:16        | ADDRMSK<7:0> |        |       |        |       |       |      | ADDR<7:0>         |       |        |       |       |              |             | 0000          |      |
|                           |                       | 15:0         | UTXISEL<1:0> | UTXINV | URXEN | UTXBRK | UTXEN | UTXBF | TRMT | URXISEL<1:0>      | ADDEN | RIDLE  | PERR  | FERR  | OERR         | URXDA       | 0110          |      |
| 8220                      | U2TXREG               | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | —            | —           | —             | 0000 |
|                           |                       | 15:0         | —            | —      | —     | —      | —     | —     | TX8  | Transmit Register |       |        |       |       |              |             | 0000          |      |
| 8230                      | U2RXREG               | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | —            | —           | —             | 0000 |
|                           |                       | 15:0         | —            | —      | —     | —      | —     | —     | RX8  | Receive Register  |       |        |       |       |              |             | 0000          |      |
| 8240                      | U2BRG <sup>(1)</sup>  | 31:16        | —            | —      | —     | —      | —     | —     | —    | —                 | —     | —      | —     | —     | BRG<19:16>   |             | 0000          |      |
|                           |                       | 15:0         | BRG<15:0>    |        |       |        |       |       |      |                   |       |        |       |       |              |             | 0000          |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** This register has corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 "CLR, SET, and INV Registers"](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 19-1: UxMODE: UARTx MODE REGISTER (“x” = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2             | Bit 25/17/9/1           | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------------------|-------------------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0                        | U-0                     | U-0           |
|           | —              | —              | —              | —              | —              | —                          | —                       | —             |
| 23:16     | R/W-0          | R-0, HS, HC    | U-0            | U-0            | U-0            | R/W-0                      | R/W-0                   | R/W-0         |
|           | SLPEN          | CLKRDY         | —              | —              | —              | CLKSEL<1:0> <sup>(1)</sup> |                         | RUNOV         |
| 15:8      | R/W-0          | U-0            | R/W-0          | R/W-0          | R/W-0          | U-0                        | R/W-0                   | R/W-0         |
|           | ON             | —              | SIDL           | IREN           | RTSMD          | —                          | UEN<1:0> <sup>(2)</sup> |               |
| 7:0       | R-0, HC        | R/W-0          | R/W-0, HC      | R/W-0          | R/W-0          | R/W-0                      | R/W-0                   | R/W-0         |
|           | WAKE           | LPBACK         | ABAUD          | RXINV          | BRGH           | PDSEL<1:0>                 |                         | STSEL         |

|                   |                      |  |
|-------------------|----------------------|--|
| <b>Legend:</b>    | HS = Set by hardware | HC = cleared by hardware                     |
| R = Readable bit  | W = Writable bit     | U = Unimplemented bit, read as ‘0’           |
| -n = Value at POR | ‘1’ = Bit is set     | ‘0’ = Bit is cleared      x = Bit is unknown |

bit 31-24 **Unimplemented:** Read as ‘0’

bit 23 **SLPEN:** Run During Sleep Enable bit

- 1 = BRG clock runs during Sleep mode
- 0 = BRG clock is turned off during Sleep mode

**Note:** SLPEN = 1 only applies if CLKSEL = FRC, or in some cases REFCLK depending on the selected REFCLK input source if running while in Sleep mode.

bit 22 **CLKRDY:** USART Clock Status bit

- 1 = UART clock is ready (User *should not* update the UxMODE register)
- 0 = UART clock is not ready (User *can* update the UxMODE register)

bit 21-19 **Unimplemented:** Read as ‘0’

bit 18-17 **CLKSEL<1:0>:** UART Baud Rate Generator Clock Selection bits<sup>(1)</sup>

- 11 = BRG clock is REFCLK1
- 10 = BRG clock is FRC
- 01 = BRG clock is SYSCLK (off in Sleep mode)
- 00 = BRG clock is PBCLKx (off in Sleep mode)

bit 16 **RUNOV:** Run During Overflow Mode bit

- 1 = Shift register continues to run when Overflow (OERR) condition is detected
- 0 = Shift register stops accepting new data when Overflow (OERR) condition is detected

bit 15 **ON:** UARTx Enable bit

- 1 = UARTx is enabled. UARTx pins are controlled by UARTx as defined by UEN<1:0> and UTXEN control bits
- 0 = UARTx is disabled. All UARTx pins are controlled by corresponding bits in the PORTx, TRISx and LATx registers; UARTx power consumption is minimal

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **SIDL:** Stop in Idle Mode bit

- 1 = Discontinue operation when device enters Idle mode
- 0 = Continue operation in Idle mode

bit 12 **IREN:** IrDA Encoder and Decoder Enable bit

- 1 = IrDA is enabled
- 0 = IrDA is disabled

**Note 1:** These bits can be changed only when the ON bit (UxMODE<15>) is set to ‘0’.

**Note 2:** These bits are present for legacy compatibility, and are superseded by PPS functionality on these devices (see [10.3 “Peripheral Pin Select \(PPS\)”](#) for more information).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 19-1: UxMODE: UARTx MODE REGISTER (CONTINUED)(“x” = 1-2)

- bit 11 **RTSMD:** Mode Selection for  $\overline{\text{UxRTS}}$  Pin bit  
1 =  $\overline{\text{UxRTS}}$  pin is in Simplex mode  
0 =  $\overline{\text{UxRTS}}$  pin is in Flow Control mode
- bit 10 **Unimplemented:** Read as ‘0’
- bit 9-8 **UEN<1:0>:** UARTx Enable bits<sup>(2)</sup>  
11 = UxTX, UxRX and UxBCLK pins are enabled and used;  $\overline{\text{UxCTS}}$  pin is controlled by corresponding bits in the PORTx register  
10 = UxTX, UxRX,  $\overline{\text{UxCTS}}$  and  $\overline{\text{UxRTS}}$  pins are enabled and used  
01 = UxTX, UxRX and  $\overline{\text{UxRTS}}$  pins are enabled and used;  $\overline{\text{UxCTS}}$  pin is controlled by corresponding bits in the PORTx register  
00 = UxTX and UxRX pins are enabled and used;  $\overline{\text{UxCTS}}$  and  $\overline{\text{UxRTS/UxBCLK}}$  pins are controlled by corresponding bits in the PORTx register
- bit 7 **WAKE:** Enable Wake-up on Start bit Detect During Sleep Mode bit  
1 = Wake-up is enabled  
0 = Wake-up is disabled
- bit 6 **LPBACK:** UARTx Loopback Mode Select bit  
1 = Loopback mode is enabled  
0 = Loopback mode is disabled
- bit 5 **ABAUD:** Auto-Baud Enable bit  
1 = Enable baud rate measurement on the next reception of Sync character (0x55); cleared by hardware upon completion  
0 = Baud rate measurement disabled or completed
- bit 4 **RXINV:** Receive Polarity Inversion bit  
1 = UxRX Idle state is ‘0’  
0 = UxRX Idle state is ‘1’
- bit 3 **BRGH:** High Baud Rate Enable bit  
1 = High-Speed mode – 4x baud clock enabled  
0 = Standard Speed mode – 16x baud clock enabled  
**Note:** Only use BRGH = 1 for Baud rates  $\geq 7.5\text{MBPS}$ .
- bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits  
11 = 9-bit data, no parity  
10 = 8-bit data, odd parity  
01 = 8-bit data, even parity  
00 = 8-bit data, no parity
- bit 0 **STSEL:** Stop Selection bit  
1 = 2 Stop bits  
0 = 1 Stop bit

**Note 1:** These bits can be changed only when the ON bit (UxMODE<15>) is set to ‘0’.

**Note 2:** These bits are present for legacy compatibility, and are superseded by PPS functionality on these devices (see [10.3 “Peripheral Pin Select \(PPS\)”](#) for more information).

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 19-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (“X” = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2       | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0                | R/W-0         | R/W-0         |
| MASK<7:0> |                |                |                |                |                |                      |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0                | R/W-0         | R/W-0         |
| ADDR<7:0> |                |                |                |                |                |                      |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0, HC      | R/W-0                | R-0           | R-1           |
|           | UTXISEL<1:0>   |                | UTXINV         | URXEN          | UTXBRK         | UTXEN <sup>(1)</sup> | UTXBF         | TRMT          |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R-1            | R-0            | R-0                  | R/W-0, HS     | R-0           |
|           | URXISEL<1:0>   |                | ADDEN          | RIDLE          | PERR           | FERR                 | OERR          | URXDA         |

|                   |                      |  |
|-------------------|----------------------|--|
| <b>Legend:</b>    | HS = Set by hardware | HC = cleared by hardware                     |
| R = Readable bit  | W = Writable bit     | U = Unimplemented bit, read as ‘0’           |
| -n = Value at POR | ‘1’ = Bit is set     | ‘0’ = Bit is cleared      x = Bit is unknown |

bit 31-24 **MASK<7:0>**: Address Match Mask bits

These bits are used to mask the ADDR<7:0> bits.

11111111 = Corresponding matching ADDR<7:0> bits are used to detect the address match

**Note:** This setting allows the user to assign individual address as well as a group broadcast address to a UART.

00000000 = Corresponding ADDR<sub>x</sub> bits are not used to detect the address match.

See [19.2 “UART Broadcast Mode Example”](#) for additional information.

bit 23-16 **ADDR<7:0>**: Automatic Address Mask bits

1 = Corresponding MASK<sub>x</sub> bits are used to detect the address match.

**Note:** This setting allows the user to assign individual address as well as a group broadcast address to a UART.

0 = Corresponding MASK<sub>x</sub> bits are not used to detect the address match.

See [19.2 “UART Broadcast Mode Example”](#) for additional information.

bit 15-14 **UTXISEL<1:0>**: TX Interrupt Mode Selection bits

11 = Reserved, do not use

10 = Interrupt is generated and asserted while the transmit buffer is empty

01 = Interrupt is generated and asserted when all characters have been transmitted

00 = Interrupt is generated and asserted while the transmit buffer contains at least one empty space

bit 13 **UTXINV**: Transmit Polarity Inversion bit

If IrDA mode is disabled (i.e., IREN (UxMODE<12>) is ‘0’):

1 = UxTX Idle state is ‘0’

0 = UxTX Idle state is ‘1’

If IrDA mode is enabled (i.e., IREN (UxMODE<12>) is ‘1’):

1 = IrDA encoded UxTX Idle state is ‘1’

0 = IrDA encoded UxTX Idle state is ‘0’

bit 12 **URXEN**: Receiver Enable bit

1 = UART<sub>x</sub> receiver is enabled. UxRX pin is controlled by UART<sub>x</sub> (if ON bit (UxMODE<15>) = 1)

0 = UART<sub>x</sub> receiver is disabled. UxRX pin is ignored by the UART<sub>x</sub> module and released to the PORT

**Note:** The event of disabling an enabled receiver will release the RX pin to the PORT function; however, the receive buffers will not be reset. Disabling the receiver has no effect on the receive status flags.

**Note 1:** This bit should not be enabled until after the ON bit (UxMODE<15>) = 1. If TX interrupts are enabled, setting this bit will immediately cause a TX interrupt based on the value of the UTXISEL bit.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 19-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED) (“X” = 1-2)

- bit 11 **UTXBRK**: Transmit Break bit  
1 = Send Break on next transmission. Start bit followed by twelve ‘0’ bits, followed by Stop bit; cleared by hardware upon completion  
0 = Break transmission is disabled or completed
- bit 10 **UTXEN**: Transmit Enable bit<sup>(1)</sup>  
1 = UARTx transmitter is enabled. UxTX pin is controlled by UARTx (if ON bit (UxMODE<15>) = 1)  
0 = UARTx transmitter is disabled  
The event of disabling an enabled transmitter will release the TX pin to the PORT function and reset the transmit buffers to empty. Any pending transmission is aborted and data characters in the transmit buffers are lost. All transmit status flags are cleared and the TRMT bit is set.
- bit 9 **UTXBF**: Transmit Buffer Full Status bit (read-only)  
1 = Transmit buffer is full  
0 = Transmit buffer is not full, at least one more character can be written
- bit 8 **TRMT**: Transmit Shift Register is Empty bit (read-only)  
1 = Transmit shift register is empty and transmit buffer is empty (the last transmission has completed)  
0 = Transmit shift register is not empty, a transmission is in progress or queued in the transmit buffer
- bit 7-6 **URXISEL<1:0>**: Receive Interrupt Mode Selection bit  
11 = Reserved  
10 = Interrupt flag bit is asserted while receive buffer is 3/4 or more full  
01 = Interrupt flag bit is asserted while receive buffer is 1/2 or more full  
00 = Interrupt flag bit is asserted while receive buffer is not empty (i.e., has at least 1 data character)
- bit 5 **ADDEN**: Address Character Detect bit (bit 8 of received data = 1)  
1 = Address Detect mode is enabled. If 9-bit mode is not selected, this control bit has no effect  
0 = Address Detect mode is disabled
- bit 4 **RIDLE**: Receiver Idle bit (read-only)  
1 = Receiver is Idle  
0 = Data is being received
- bit 3 **PERR**: Parity Error Status bit (read-only)  
1 = Parity error has been detected for the current character  
0 = Parity error has not been detected
- bit 2 **FERR**: Framing Error Status bit (read-only)  
1 = Framing error has been detected for the current character  
0 = Framing error has not been detected
- bit 1 **OERR**: Receive Buffer Overrun Error Status bit.  
When RUNOV = 0, clearing a previously set OERR bit will clear and reset the receive buffer and shift register.  
When RUNOV = 1, Clearing a previously set OERR bit will NOT reset the receive buffer and shift register  
1 = Receive buffer has overflowed  
0 = Receive buffer has not overflowed
- bit 0 **URXDA**: Receive Buffer Data Available bit (read-only)  
1 = Receive buffer has data, at least one more character can be read  
0 = Receive buffer is empty

**Note 1:** This bit should not be enabled until after the ON bit (UxMODE<15>) = 1. If TX interrupts are enabled, setting this bit will immediately cause a TX interrupt based on the value of the UTXISEL bit.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 19-3: UxRXREG: UARTx RECEIVE REGISTER (“X” = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R-0           |
|           | —              | —              | —              | —              | —              | —              | —             | RX<8>         |
| 7:0       | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | RX<7:0>        |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as ‘0’  
 -n = Value at POR                      ‘1’ = Bit is set                      ‘0’ = Bit is cleared                      x = Bit is unknown

- bit 31-9 **Unimplemented:** Read as ‘0’
- bit 8 **RX<8>:** Data bit 8 of the received character (in 9-bit mode)
- bit 7-0 **RX<7:0>:** Data bits 7-0 of the received character

**REGISTER 19-4: UxTXREG: UARTx TRANSMIT REGISTER (“X” = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-x            | U-x            | U-x            | U-x            | U-x            | U-x            | U-x           | U-x           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-x            | U-x            | U-x            | U-x            | U-x            | U-x            | U-x           | U-x           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-x            | U-x            | U-x            | U-x            | U-x            | U-x            | U-x           | W-x           |
|           | —              | —              | —              | —              | —              | —              | —             | TX<8>         |
| 7:0       | W-0            | W-0            | W-0            | W-0            | W-0            | W-0            | W-0           | W-0           |
|           | TX<7:0>        |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as ‘0’  
 -n = Value at POR                      ‘1’ = Bit is set                      ‘0’ = Bit is cleared                      x = Bit is unknown

- bit 31-9 **Unimplemented:** Read as initialized data
- bit 8 **TX<8>:** Data bit 8 of the transmitted character (in 9-bit mode)
- bit 7-0 **TX<7:0>:** Data bits 7-0 of the transmitted character

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 19-5: UxBRG: UARTx BAUD RATE GENERATOR REGISTER (“x” = 1-2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | BRG<19:16>     |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | BRG<15:8>      |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | BRG<7:0>       |                |                |                |                |                |               |               |

**Legend:**  
R = Readable bit                                    W = Writable bit                                    U = Unimplemented bit, read as '0'  
-n = Value at POR                                    '1' = Bit is set                                    '0' = Bit is cleared                                    x = Bit is unknown

bit 31-20 **Unimplemented:** Read as '0'  
bit 19-0 **BRG<19:0>:** Baud Rate Generator Divisor bits  
**Note:** The UxBRG register cannot be changed while UARTx is enabled (ON bit (UxMODE<15>) = 1)).

**TABLE 19-2: UART BAUD RATE CALCULATIONS**

| UART Baud Rate With | UxBRG Equals   |
|---------------------|--|
| BRGH = 0            | $UxBRG = ((CLKSEL \text{ Frequency} / (16 * \text{Desired Baud Rate})) - 1)$ |
| BRGH = 1            | $UxBRG = ((CLKSEL \text{ Frequency} / (4 * \text{Desired Baud Rate})) - 1)$  |

**Note:** UART1 and UART2 on PBCLK2.

# PIC32MK GPG/MCJ with CAN FD Family

## 19.2 UART Broadcast Mode Example

As shown in Table 19-3, the group hardware address identifier bit was arbitrarily chosen as bit 7 with bit 4 chosen as the software group or individual UART target ID. Therefore, the collective group address assigned for all UARTs (i.e, [w, x, y, z]) is `'0b100100xx`, while the individual addresses are `'0b10000000` through `'0b10000011`, respectively.

Any MASK register bit = 0 means the corresponding ADDR<7:0> bit is a “don't care” from a hardware address matching point of view. Using this scheme, multiple UART subnet groups could be created within a network. If not using address match with a broadcast mode, set the ADDRMSK<7:0> bits (UxSTAT<31:24>) = 0x00, which is the default.

To send a broadcast message to all UARTs in the group identified by bit 7 = 1, send UxTXREG = (0x190), address bit 9 set. All the UARTs in that group, bit 7 = 1, would generate an interrupt for an address match because of the bit <7:5>, <3:2> match, Logic AND of MASK and ADDR registers equal “true”. User software would check if bit 4 = 1, and if true, the RX<7:0> bits register value is valid for all UARTS.

To send a specific message to UARTy within the group, the user would send UxTXREG = (0x182), address bit 9 set. All of the UARTs in that group identified with bit 7 = 1 would still generate an interrupt for an address match because of the bit <7:5>, <3:2> address match, Logic AND of MASK and ADDR registers equal True. In this case, user software would check if bit 4 = 0, and if true, the RX<7:0> bits register value would be intended only for UARTy, with all others ignored.

**TABLE 19-3: PDSEL<1:0> (UxMODE<2:1>) = '0b11 AND ADM\_EN (UxSTA<24>) = 1**

| Networked UARTS | Register Bit | 7 | 6 | 5 | 4                           | 3 | 2 | 1 | 0 | Individual/ Group Addresses |
|-----------------|--------------|---|---|---|-----------------------------|---|---|---|---|-----------------------------|
| UARTx           | ADDRMSK      | 1 | 1 | 1 | 0                           | 1 | 1 | 0 | 0 | 0xBC                        |
| UARTw           | ADDR         | 1 | 0 | 0 | 1 = Group<br>0 = Individual | 0 | 0 | 0 | 0 | 0x80/0x9X                   |
| UARTx           | ADDR         | 1 | 0 | 0 | 1 = Group<br>0 = Individual | 0 | 0 | 0 | 1 | 0x81/0x9X                   |
| UARTy           | ADDR         | 1 | 0 | 0 | 1 = Group<br>0 = Individual | 0 | 0 | 1 | 0 | 0x82/0x9X                   |
| UARTz           | ADDR         | 1 | 0 | 0 | 1 = Group<br>0 = Individual | 0 | 0 | 1 | 1 | 0x83/0x9X                   |

# PIC32MK GPG/MCJ with CAN FD Family

---

## 19.3 Module Operation

### 19.3.1 INITIALIZATION

Clearing the ON bit (i.e., = 0), which disables the UART module, will do the following:

- Aborts all pending transmissions and receptions and resets the module, as follows:
  - Reset the RX/TX buffers/FIFO to empty states (any data characters in the buffers are lost)
  - Resets the baud rate counter (UxBRG is not affected, only the counter)
  - Resets all error and status flags: URXDA, OERR, FERR, PERR, UTXBRK, UTXBF are cleared and RIDLE, TRMT are set
- Stop clocks to the entire module with the exception of the SFRs, saving power
- Surrenders control of the module I/O pins

**Note:** Once the ON bit is set, it should not be cleared until the CLKRDY bit is read to be a logic '1'. This allows proper synchronization of the status and output signals. Otherwise, glitches in the status signals or BRG clock can occur.

Setting the ON bit (i.e., = 1), which enables the UART module, will do the following:

- The UART module controls the I/O pins as defined by the UEN bits, overriding the port TRIS and LATCH register bit settings
- UxTX is forced as an output driving the idle state defined by the UTXINV bit, when no transmissions are taking place
- UxRX is configured as an input
- If CTS and RTS are enabled, CTS is forced as an input and the RTS/BCLK pin functions as RTS output
- If BCLK is enabled, the RTS/BCLK output drives the 16x baud clock output

**Note:** The ON bit should not be set (i.e., = 1) unless the CLKRDY bit is read to be a logic '0'.

## 19.4 Serial Protocols Usage

### 19.4.1 DATA TERMINAL EQUIPMENT (DTE) WITH FLOW CONTROL

When connecting to the DTE (typically a PC) and flow control is desired, set the UEN bit = 10 to enable CTS and RTS, and set the RTSMD bit = 0.

### 19.4.2 IEEE-485

To use the UART module in the IEEE-485 protocol, use the address detection feature to detect message frames. Normally, set the UEN bit = '01' to drive the RTS pin and control the bus driver, and set the RTSMD bit = 1.

### 19.4.3 LIN BUS

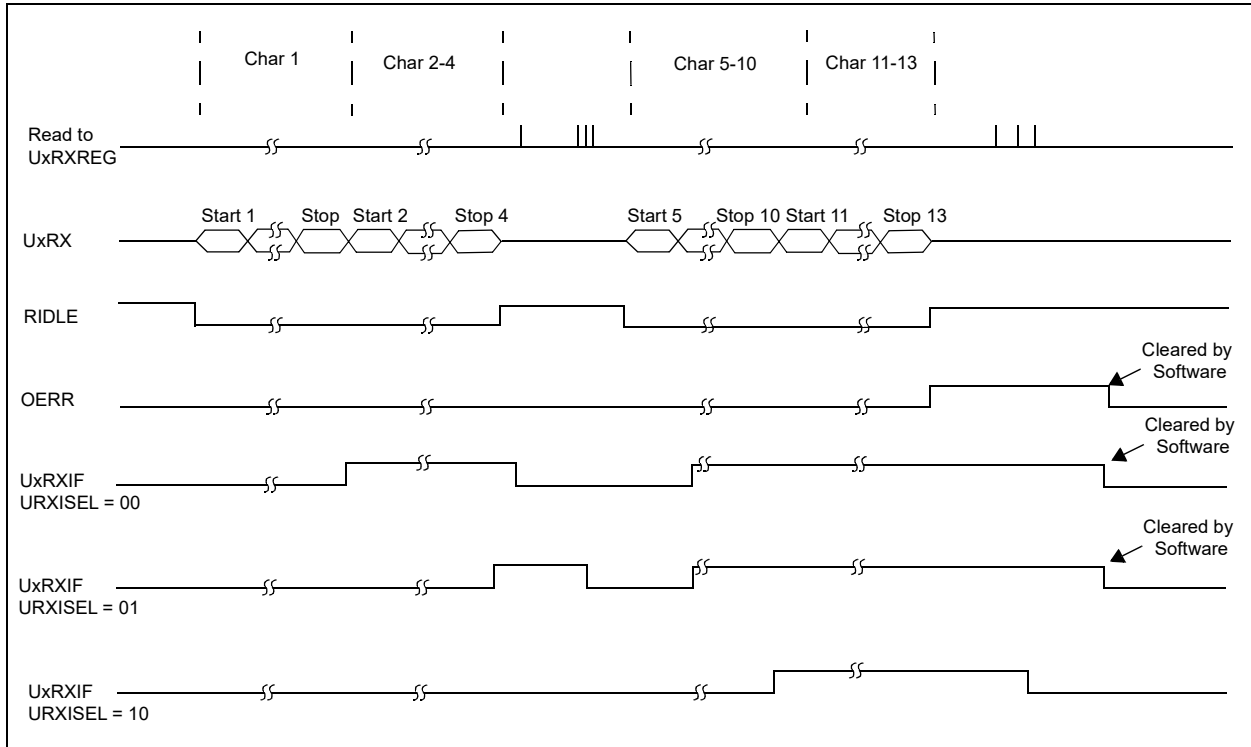
To transmit on a LIN bus, the transmitter must send a frame in 8,N,1 format consisting of a break, a synchronization character (0x55), and the message body. The module has extensive support for the LIN protocol including bus wake-up for a slave node as well as auto-baud detection and BREAK character transmit for master nodes. When in LIN mode, the software should program the BRGH bit = 0, which insures a 16x baud clock is used with majority detect.

# PIC32MK GPG/MCJ with CAN FD Family

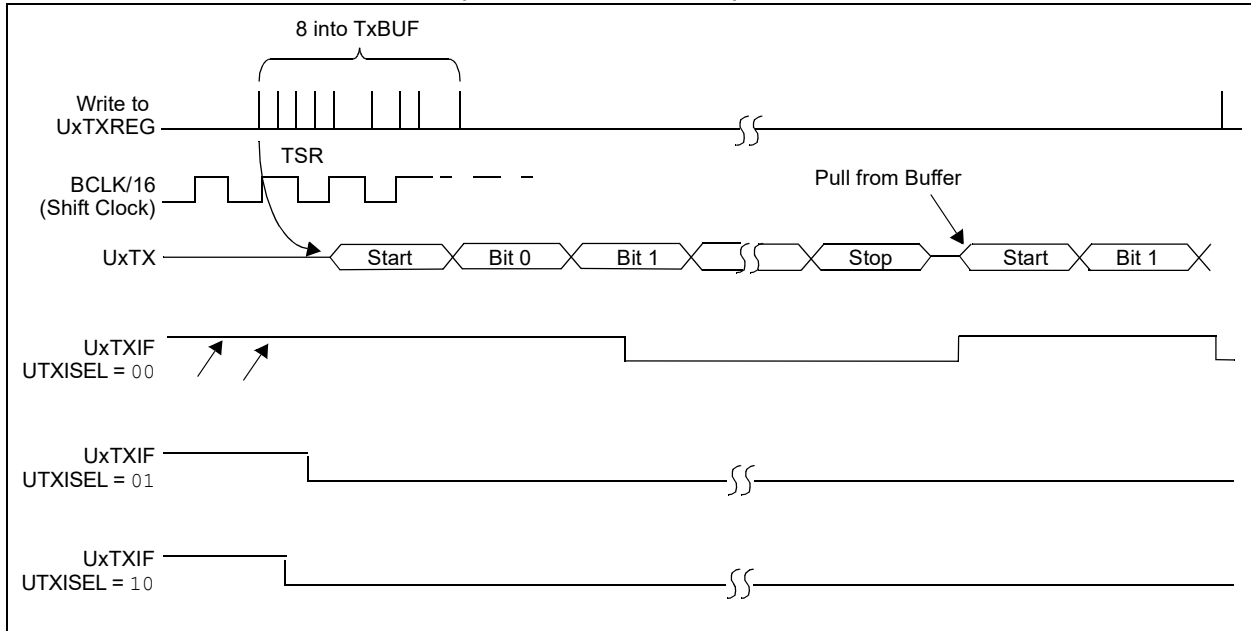
## 19.5 Transmit and Receive Timing

Figure 19-2 and Figure 19-3 illustrate typical receive and transmit timing for the UART module.

**FIGURE 19-2: UART RECEPTION**



**FIGURE 19-3: TRANSMISSION (8-BIT OR 9-BIT DATA)**



# PIC32MK GPG/MCJ with CAN FD Family

---

NOTES:

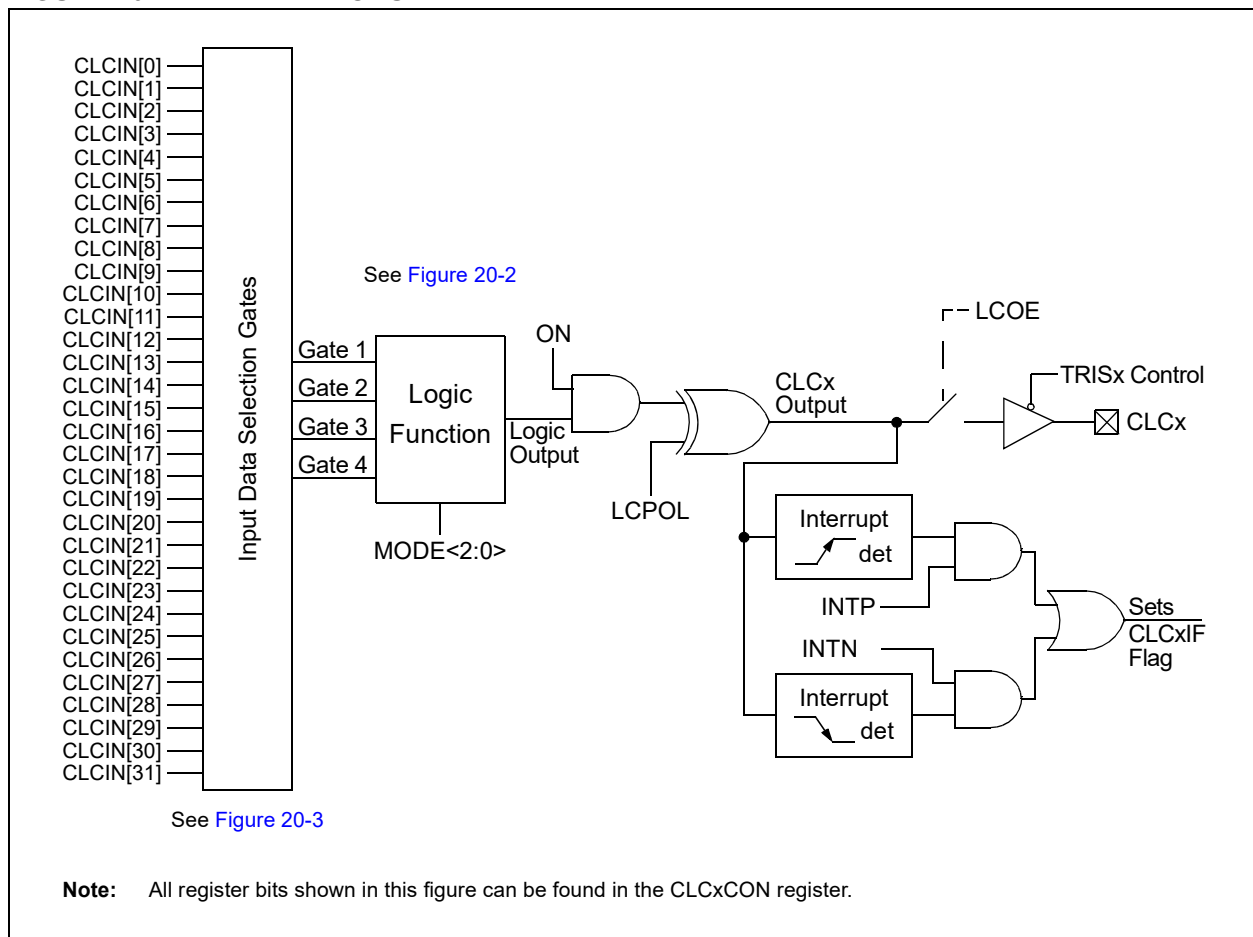
## 20.0 CONFIGURABLE LOGIC CELL (CLC)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 36. “Configurable Logic Cell”** (DS60001363) in the “PIC32 Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)). The information in this data sheet supersedes the information in the FRM.

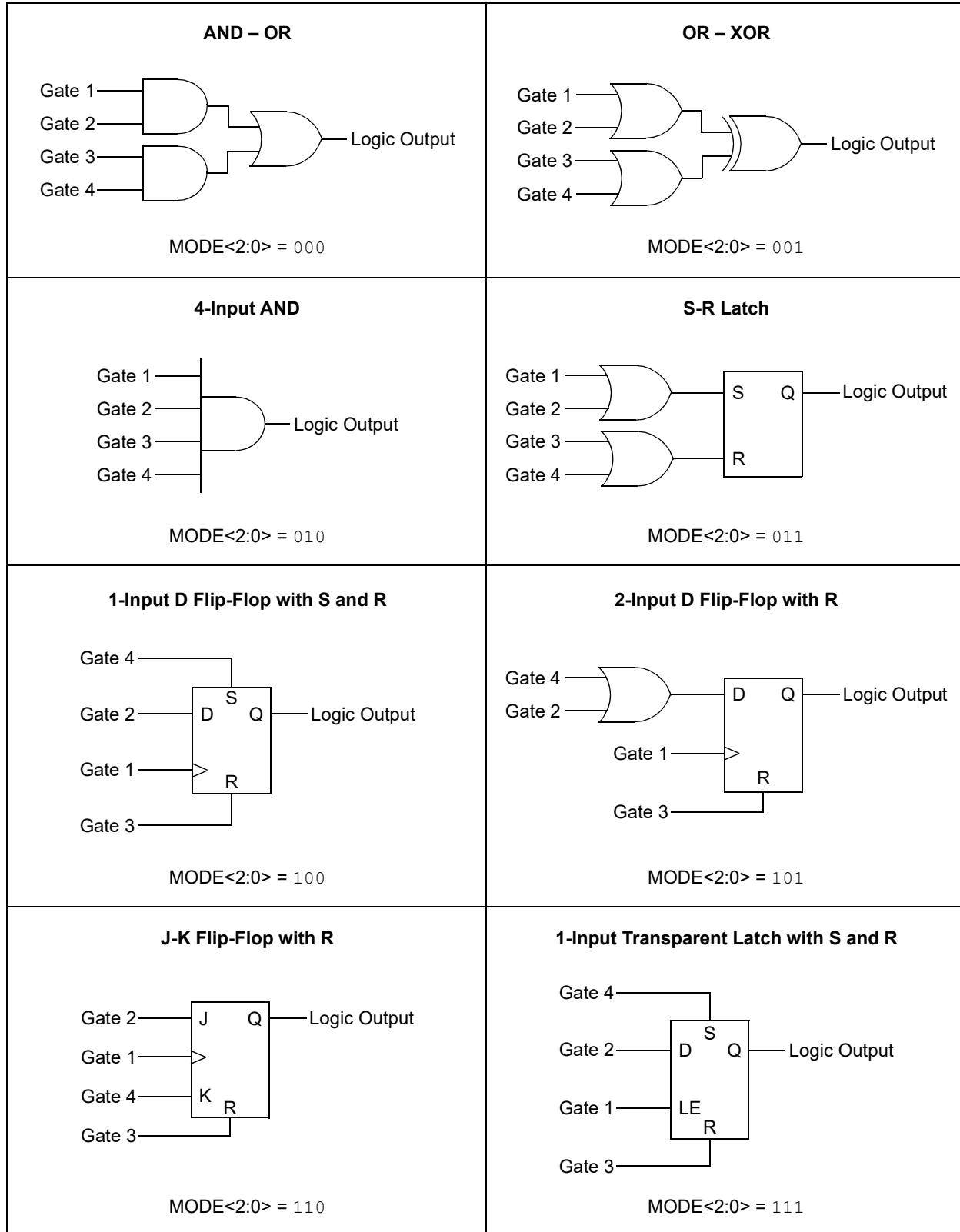
The Configurable Logic Cell (CLC) module allows the user to specify combinations of signals as inputs to a logic function and to use the logic output to control other peripherals or I/O pins. This provides greater flexibility and potential in embedded designs since the CLC module can operate outside the limitations of software execution, and supports a vast amount of output designs.

There are four input gates to the selected logic function. These four input gates select from a pool of up to 32 signals that are selected using four data source selection multiplexers. Figure 20-1 shows an overview of the module. Figure 20-3 shows the details of the data source multiplexers and logic input gate connections.

**FIGURE 20-1: CLCx MODULE**

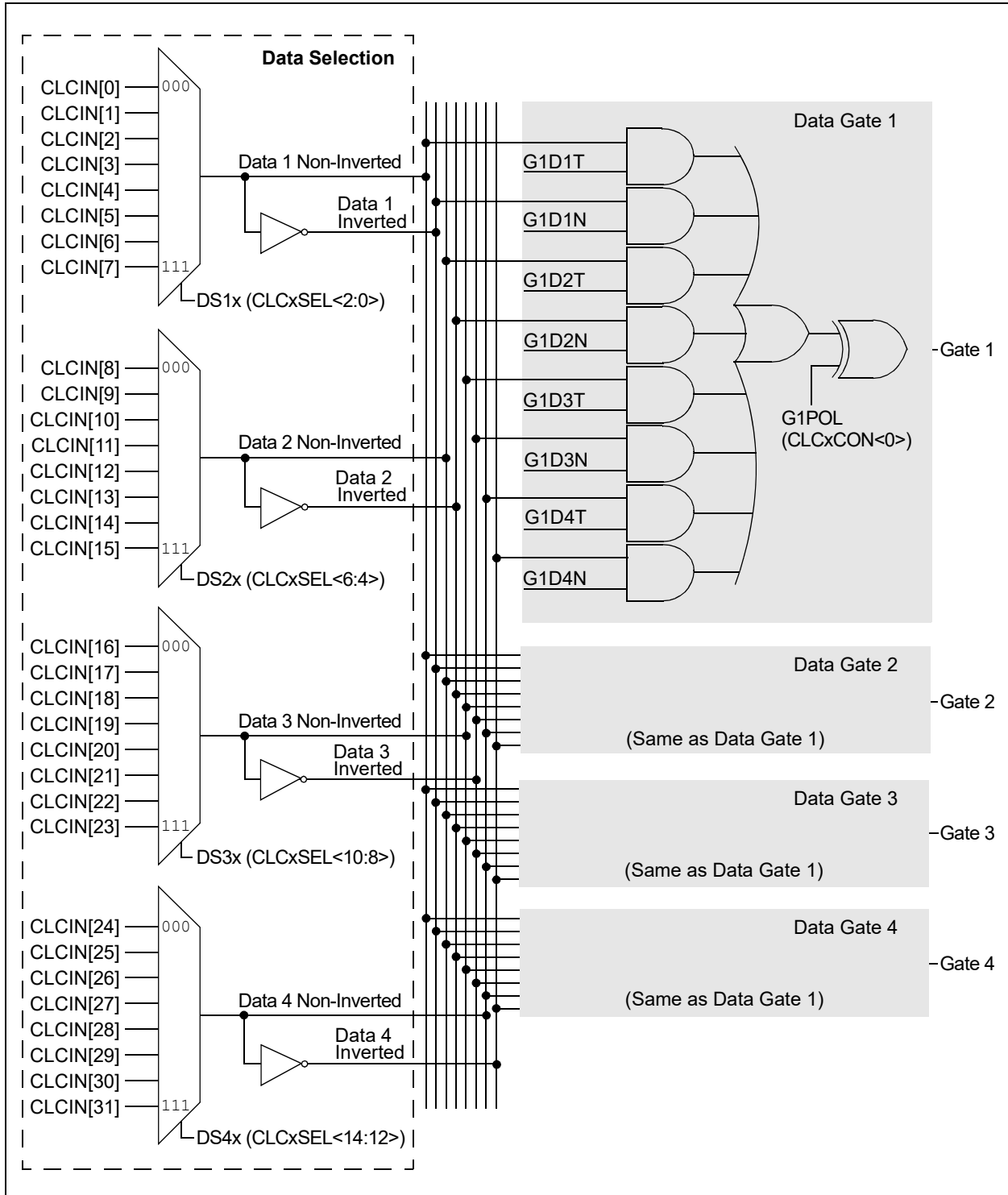


**FIGURE 20-2: CLCx LOGIC FUNCTION COMBINATORIAL OPTIONS**





**FIGURE 20-3: CLCx INPUT SOURCE SELECTION DIAGRAM**



---

---

## 20.1 Control Registers

The CLCx module is controlled by the following registers:

- CLCxCON
- CLCxSEL
- CLCxGLS

The CLCx Control register (CLCxCON) is used to enable the module and interrupts, control the output enable bit, select output polarity and select the logic function. The CLCx Control registers also allow the user to control the logic polarity of not only the cell output, but also some intermediate variables.

The CLCx Source Select register (CLCxSEL) allows the user to select up to 4 data input sources using the 4 data input selection multiplexers. Each multiplexer has a list of 8 data sources available.

The CLCx Gate Logic Select register (CLCxGLS) allows the user to select which outputs from each of the selection MUXes are used as inputs to the input gates of the logic cell. Each data source MUX outputs both a true and a negated version of its output. All of these 8 signals are enabled, ORed together by the logic cell input gates.

**TABLE 20-1: CLC1, CLC2 AND CLC3 REGISTER MAP**

| Virtual Address (BF80_#) | Register Name <sup>(1)</sup> | Bit Range | Bits  |          |       |       |       |          |       |       |       |          |       |       |       |          |           | All Resets |       |      |
|--------------------------|------------------------------|-----------|-------|----------|-------|-------|-------|----------|-------|-------|-------|----------|-------|-------|-------|----------|-----------|------------|-------|------|
|                          |                              |           | 31/15 | 30/14    | 29/13 | 28/12 | 27/11 | 26/10    | 25/9  | 24/8  | 23/7  | 22/6     | 21/5  | 20/4  | 19/3  | 18/2     | 17/1      |            | 16/0  |      |
| 2480                     | CLC1CON                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | G4POL    | G3POL     | G2POL      | G1POL | 0000 |
|                          |                              | 15:0      | ON    | —        | —     | —     | —     | INTP     | INTN  | —     | —     | LCOE     | LCOUT | LCPOL | —     | —        | MODE<2:0> |            |       | 0000 |
| 2490                     | CLC1SEL                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —         | —          | —     | 0000 |
|                          |                              | 15:0      | —     | DS4<2:0> |       |       | —     | DS3<2:0> |       |       | —     | DS2<2:0> |       |       | —     | DS1<2:0> |           |            | 0000  |      |
| 24A0                     | CLC1GLS                      | 32:16     | G4D4T | G4D4N    | G4D3T | G4D3N | G4D2T | G4D2N    | G4D1T | G4D1N | G3D4T | G3D4N    | G3D3T | G3D3N | G3D2T | G3D2N    | G3D1T     | G3D1N      | 0000  |      |
|                          |                              | 15:0      | G2D4T | G2D4N    | G2D3T | G2D3N | G2D2T | G2D2N    | G2D1T | G2D1N | G1D4T | G1D4N    | G1D3T | G1D3N | G1D2T | G1D2N    | G1D1T     | G1D1N      | 0000  |      |
| 2500                     | CLC2CON                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | G4POL    | G3POL     | G2POL      | G1POL | 0000 |
|                          |                              | 15:0      | ON    | —        | —     | —     | —     | INTP     | INTN  | —     | —     | LCOE     | LCOUT | LCPOL | —     | —        | MODE<2:0> |            |       | 0000 |
| 2510                     | CLC2SEL                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —         | —          | —     | 0000 |
|                          |                              | 15:0      | —     | DS4<2:0> |       |       | —     | DS3<2:0> |       |       | —     | DS2<2:0> |       |       | —     | DS1<2:0> |           |            | 0000  |      |
| 2520                     | CLC2GLS                      | 32:16     | G4D4T | G4D4N    | G4D3T | G4D3N | G4D2T | G4D2N    | G4D1T | G4D1N | G3D4T | G3D4N    | G3D3T | G3D3N | G3D2T | G3D2N    | G3D1T     | G3D1N      | 0000  |      |
|                          |                              | 15:0      | G2D4T | G2D4N    | G2D3T | G2D3N | G2D2T | G2D2N    | G2D1T | G2D1N | G1D4T | G1D4N    | G1D3T | G1D3N | G1D2T | G1D2N    | G1D1T     | G1D1N      | 0000  |      |
| 2580                     | CLC3CON                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | G4POL    | G3POL     | G2POL      | G1POL | 0000 |
|                          |                              | 15:0      | ON    | —        | —     | —     | —     | INTP     | INTN  | —     | —     | LCOE     | LCOUT | LCPOL | —     | —        | MODE<2:0> |            |       | 0000 |
| 2590                     | CLC3SEL                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —         | —          | —     | 0000 |
|                          |                              | 15:0      | —     | DS4<2:0> |       |       | —     | DS3<2:0> |       |       | —     | DS2<2:0> |       |       | —     | DS1<2:0> |           |            | 0000  |      |
| 25A0                     | CLC3GLS                      | 32:16     | G4D4T | G4D4N    | G4D3T | G4D3N | G4D2T | G4D2N    | G4D1T | G4D1N | G3D4T | G3D4N    | G3D3T | G3D3N | G3D2T | G3D2N    | G3D1T     | G3D1N      | 0000  |      |
|                          |                              | 15:0      | G2D4T | G2D4N    | G2D3T | G2D3N | G2D2T | G2D2N    | G2D1T | G2D1N | G1D4T | G1D4N    | G1D3T | G1D3N | G1D2T | G1D2N    | G1D1T     | G1D1N      | 0000  |      |
| 2600                     | CLC4CON                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | G4POL    | G3POL     | G2POL      | G1POL | 0000 |
|                          |                              | 15:0      | ON    | —        | —     | —     | —     | INTP     | INTN  | —     | —     | LCOE     | LCOUT | LCPOL | —     | —        | MODE<2:0> |            |       | 0000 |
| 2610                     | CLC4SEL                      | 32:16     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —     | —     | —     | —        | —         | —          | —     | 0000 |
|                          |                              | 15:0      | —     | DS4<2:0> |       |       | —     | DS3<2:0> |       |       | —     | DS2<2:0> |       |       | —     | DS1<2:0> |           |            | 0000  |      |
| 2620                     | CLC4GLS                      | 32:16     | G4D4T | G4D4N    | G4D3T | G4D3N | G4D2T | G4D2N    | G4D1T | G4D1N | G3D4T | G3D4N    | G3D3T | G3D3N | G3D2T | G3D2N    | G3D1T     | G3D1N      | 0000  |      |
|                          |                              | 15:0      | G2D4T | G2D4N    | G2D3T | G2D3N | G2D2T | G2D2N    | G2D1T | G2D1N | G1D4T | G1D4N    | G1D3T | G1D3N | G1D2T | G1D2N    | G1D1T     | G1D1N      | 0000  |      |

**Legend:** — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET and INV registers at their virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively.

## REGISTER 20-1: CLCxCON: CLCx CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3      | Bit 26/18/10/2      | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|---------------------|---------------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0                 | U-0                 | U-0           | U-0           |
|           | —              | —              | —              | —              | —                   | —                   | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | R/W-0               | R/W-0               | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | G4POL               | G3POL               | G2POL         | G1POL         |
| 15:8      | R/W-0          | U-0            | U-0            | U-0            | R/W-0               | R/W-0               | U-0           | U-0           |
|           | ON             | —              | —              | —              | INTP <sup>(1)</sup> | INTN <sup>(1)</sup> | —             | —             |
| 7:0       | R/W-0          | R-0, HS, HC    | R/W-0          | U-0            | U-0                 | R/W-0               | R/W-0         | R/W-0         |
|           | LCOE           | LCOUT          | LCPOL          | —              | —                   | MODE<2:0>           |               |               |

|                   |                             |  |
|-------------------|-----------------------------|--|
| <b>Legend:</b>    | HC = Hardware Clearable bit | HS = Hardware Settable bit                   |
| R = Readable bit  | W = Writable bit            | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set            | '0' = Bit is cleared      x = Bit is unknown |

bit 31-20 **Unimplemented:** Read as '0'

bit 19 **G4POL:** Gate 4 Polarity Control bit

- 1 = The output of Channel 4 logic is inverted when applied to the logic cell
- 0 = The output of Channel 4 logic is not inverted

bit 18 **G3POL:** Gate 3 Polarity Control bit

- 1 = The output of Channel 3 logic is inverted when applied to the logic cell
- 0 = The output of Channel 3 logic is not inverted

bit 17 **G2POL:** Gate 2 Polarity Control bit

- 1 = The output of Channel 2 logic is inverted when applied to the logic cell
- 0 = The output of Channel 2 logic is not inverted

bit 16 **G1POL:** Gate 1 Polarity Control bit

- 1 = The output of Channel 1 logic is inverted when applied to the logic cell
- 0 = The output of Channel 1 logic is not inverted

bit 15 **ON:** CLCx Enable bit

- 1 = CLCx is enabled and mixing input signals
- 0 = CLCx is disabled and has logic zero outputs

bit 14-13 **Unimplemented:** Read as '0'

bit 12 **Unimplemented:** Read as '0'

bit 11 **INTP:** CLCx Positive Edge Interrupt Enable bit<sup>(1)</sup>

- 1 = Interrupt will be generated when a rising edge occurs on LCOUT
- 0 = Interrupt will not be generated

bit 10 **INTN:** CLCx Negative Edge Interrupt Enable bit<sup>(1)</sup>

- 1 = Interrupt will be generated when a falling edge occurs on LCOUT
- 0 = Interrupt will not be generated

bit 9-8 **Unimplemented:** Read as '0'

bit 7 **LCOE:** CLCx Port Enable bit

- 1 = CLCx port pin output is enabled
- 0 = CLCx port pin output is disabled

bit 6 **LCOUT:** CLCx Data Output Status bit

- 1 = CLCx output high
- 0 = CLCx output low

**Note 1:** The INTP and INTN bits should never be set at the same time when ON bit is enabled for proper interrupt functionality.

## REGISTER 20-1: CLCxCON: CLCx CONTROL REGISTER (CONTINUED)

- bit 5     **LCPOL**: CLCx Output Polarity Control bit  
          1 = The output of the module is inverted  
          0 = The output of the module is not inverted
- bit 4-3   **Unimplemented**: Read as '0'
- bit 2-0   **MODE<2:0>**: CLCx Mode bits  
          111 = Cell is a 1-input transparent latch with S and R  
          110 = Cell is a JK flip-flop with R  
          101 = Cell is a 2-input D flip-flop with R  
          100 = Cell is a 1-input D flip-flop with S and R  
          011 = Cell is an SR latch  
          010 = Cell is a 4-input AND  
          001 = Cell is an OR-XOR  
          000 = Cell is a AND-OR

**Note 1:** The INTP and INTN bits should never be set at the same time when ON bit is enabled for proper interrupt functionality.

**REGISTER 20-2: CLCxSEL: CLCx INPUT MUX SELECT REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0            | R/W-0          | R/W-0          | R/W-0          | U-0            | R/W-0          | R/W-0         | R/W-0         |
|           | —              | DS4<2:0>       |                |                | —              | DS3<2:0>       |               |               |
| 7:0       | U-0            | R/W-0          | R/W-0          | R/W-0          | U-0            | R/W-0          | R/W-0         | R/W-0         |
|           | —              | DS2<2:0>       |                |                | —              | DS1<2:0>       |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-15 **Unimplemented:** Read as '0'

bit 14-12 **DS4<2:0>**: Data Selection MUX 4 Signal Selection bits

For CLC1:

- 000 = CLCINB Pad
- 001 = CLC2 Out
- 010 = TMR2 Sync Out
- 011 = SPI1 SDI In
- 100 = RTCC Event
- 101 = ICAP4 Event
- 110 = OCMP4 Output
- 111 = UART1 BCLK/RTS

For CLC2:

- 000 = CLCINB Pad
- 001 = CLC3 Out
- 010 = TMR3 Sync
- 011 = SPI2 SDI In
- 100 = PWM3H Out
- 101 = ICAP4 Event
- 110 = OCMP4 Output
- 111 = TMR5 Compare Event

For CLC3:

- 000 = CLCIND Pad
- 001 = CLC4 Out
- 010 = TMR6 Sync
- 011 = PWM6H Out
- 100 = RTCC Event
- 101 = ICAP8 Event
- 110 = OCMP8 Output
- 111 = ADC2 End of Conversion

For CLC4:

- 000 = CLCIND Pad
- 001 = CLC4 Out
- 010 = ADC5 End of Conversion
- 011 = PWM9H Out
- 100 = RTCC Event
- 101 = ICAP8 Event
- 110 = OCMP8 Output
- 111 = TMR9 Compare Event

## REGISTER 20-2: CLCxSEL: CLCx INPUT MUX SELECT REGISTER (CONTINUED)

- bit 11    **Unimplemented:** Read as '0'
- bit 10-8    **DS3<2:0>:** Data Selection MUX 3 Signal Selection bits
- For CLC1:
- 000 = CLC1NA Pad
  - 001 = CLC1 Out
  - 010 = CMP2 Out
  - 011 = SPI1 SDO Out
  - 100 = UART1 RX In
  - 101 = ICAP3 Event
  - 110 = OCMP3 Output
  - 111 = TMR4 Compare Event
- For CLC2:
- 000 = CLC2NA Pad
  - 001 = CLC2 Out
  - 010 = CMP3 Out
  - 011 = SPI2 SDO Out
  - 100 = PWM2H Out
  - 101 = ICAP3 Event
  - 110 = OCMP3 Output
  - 111 = TMR4 Compare Event
- For CLC3:
- 000 = CLC3NC Pad
  - 001 = CLC3 Out
  - 010 = CMP4 Out
  - 011 = PWM5H Out
  - 100 = UART2 RX In
  - 101 = ICAP7 Event
  - 110 = OCMP7 Output
  - 111 = TMR8 Compare Event
- For CLC4:
- 000 = CLC4NC Pad
  - 001 = CLC3 Out
  - 010 = CMP5 Out
  - 011 = PWM8H Out
  - 100 = ADC4 End of Conversion
  - 101 = ICAP7 Event
  - 110 = OCMP7 Output
  - 111 = TMR8 Compare Event
- bit 7    **Unimplemented:** Read as '0'

---

---

**REGISTER 20-2: CLCxSEL: CLCx INPUT MUX SELECT REGISTER (CONTINUED)**

bit 6-4 **DS2<2:0>**: Data Selection MUX 2 Signal Selection bits

For CLC1:

000 = CLCINB Pad  
001 = CLC2 Out  
010 = CMP1 Out  
011 = UART1 TX Out  
100 = ADC7 End of Conversion  
101 = ICAP2 Event  
110 = OCMP2 Output  
111 = TMR3 Compare Event

For CLC2:

000 = CLCINB Pad  
001 = PWM1H Out  
010 = CMP2 Out  
011 = UART2 TX Out  
100 = ADC0 End of Conversion  
101 = ICAP2 Event  
110 = OCMP2 Output  
111 = TMR3 Compare Event

For CLC3:

000 = CLCIND Pad  
001 = CLC4 Out  
010 = CMP3 Out  
011 = PWM4H Out  
100 = ADC1 End of Conversion  
101 = ICAP6 Event  
110 = OCMP6 Output  
111 = TMR7 Compare Event

For CLC4:

000 = CLCIND Pad  
001 = DMA Channel 0 Interrupt  
010 = CMP4 Out  
011 = PWM7H Out  
100 = ADC3 End of Conversion  
101 = ICAP6 Event  
110 = OCMP6 Output  
111 = TMR7 Compare Event

bit 3 **Unimplemented:** Read as '0'



## REGISTER 20-2: CLCxSEL: CLCx INPUT MUX SELECT REGISTER (CONTINUED)

bit 2-0 **DS1<2:0>**: Data Selection MUX 1 Signal Selection bits

For CLC1:

000 = CLCINA Pad  
001 = System Clock  
010 = SOSC Clock  
011 = LPRC Clock  
100 = REFO2 Clock Output  
101 = ICAP1 Event  
110 = OCMP1 Output  
111 = TMR2 Compare Event

For CLC2:

000 = CLCINA Pad  
001 = System Clock  
010 = SOSC Clock  
011 = LPRC Clock  
100 = REFO2 Clock Output  
101 = ICAP1 Event  
110 = OCMP1 Output  
111 = TMR2 Compare Event

For CLC3:

000 = CLCINC Pad  
001 = System Clock  
010 = SOSC Clock  
011 = LPRC Clock  
100 = REFO2 Clock Output  
101 = ICAP5 Event  
110 = OCMP5 Output  
111 = TMR6 Compare Event

For CLC4:

000 = CLCINC Pad  
001 = System Clock  
010 = SOSC Clock  
011 = LPRC Clock  
100 = REFO2 Clock Output  
101 = ICAP5 Event  
110 = OCMP5 Output  
111 = TMR6 Compare Event

**REGISTER 20-3: CLCxGLS: CLCx GATE LOGIC INPUT SELECT REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | G4D4T          | G4D4N          | G4D3T          | G4D3N          | G4D2T          | G4D2N          | G4D1T         | G4D1N         |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | G3D4T          | G3D4N          | G3D3T          | G3D3N          | G3D2T          | G3D2N          | G3D1T         | G3D1N         |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | G2D4T          | G2D4N          | G2D3T          | G2D3N          | G2D2T          | G2D2N          | G2D1T         | G2D1N         |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | G1D4T          | G1D4N          | G1D3T          | G1D3N          | G1D2T          | G1D2N          | G1D1T         | G1D1N         |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31     **G4D4T:** Gate 4 Data Source 4 True Enable bit  
1 = The Data Source 4 signal is enabled for Gate 4  
0 = The Data Source 4 signal is disabled for Gate 4
- bit 30     **G4D4N:** Gate 4 Data Source 4 Negated Enable bit  
1 = The Data Source 4 inverted signal is enabled for Gate 4  
0 = The Data Source 4 inverted signal is disabled for Gate 4
- bit 29     **G4D3T:** Gate 4 Data Source 3 True Enable bit  
1 = The Data Source 3 signal is enabled for Gate 4  
0 = The Data Source 3 signal is disabled for Gate 4
- bit 28     **G4D3N:** Gate 4 Data Source 3 Negated Enable bit  
1 = The Data Source 3 inverted signal is enabled for Gate 4  
0 = The Data Source 3 inverted signal is disabled for Gate 4
- bit 27     **G4D2T:** Gate 4 Data Source 2 True Enable bit  
1 = The Data Source 2 signal is enabled for Gate 4  
0 = The Data Source 2 signal is disabled for Gate 4
- bit 26     **G4D2N:** Gate 4 Data Source 2 Negated Enable bit  
1 = The Data Source 2 inverted signal is enabled for Gate 4  
0 = The Data Source 2 inverted signal is disabled for Gate 4
- bit 25     **G4D1T:** Gate 4 Data Source 1 True Enable bit  
1 = The Data Source 1 signal is enabled for Gate 4  
0 = The Data Source 1 signal is disabled for Gate 4
- bit 24     **G4D1N:** Gate 4 Data Source 1 Negated Enable bit  
1 = The Data Source 1 inverted signal is enabled for Gate 4  
0 = The Data Source 1 inverted signal is disabled for Gate 4
- bit 23     **G3D4T:** Gate 3 Data Source 4 True Enable bit  
1 = The Data Source 4 signal is enabled for Gate 3  
0 = The Data Source 4 signal is disabled for Gate 3
- bit 22     **G3D4N:** Gate 3 Data Source 4 Negated Enable bit  
1 = The Data Source 4 inverted signal is enabled for Gate 3  
0 = The Data Source 4 inverted signal is disabled for Gate 3
- bit 21     **G3D3T:** Gate 3 Data Source 3 True Enable bit  
1 = The Data Source 3 signal is enabled for Gate 3  
0 = The Data Source 3 signal is disabled for Gate 3

# PIC32MK GPG/MCJ FAMILY

## REGISTER 20-3: CLCxGLS: CLCx GATE LOGIC INPUT SELECT REGISTER (CONTINUED)

- bit 20    **G3D3N:** Gate 3 Data Source 3 Negated Enable bit  
1 = The Data Source 3 inverted signal is enabled for Gate 3  
0 = The Data Source 3 inverted signal is disabled for Gate 3
- bit 19    **G3D2T:** Gate 3 Data Source 2 True Enable bit  
1 = The Data Source 2 signal is enabled for Gate 3  
0 = The Data Source 2 signal is disabled for Gate 3
- bit 18    **G3D2N:** Gate 3 Data Source 2 Negated Enable bit  
1 = The Data Source 2 inverted signal is enabled for Gate 3  
0 = The Data Source 2 inverted signal is disabled for Gate 3
- bit 17    **G3D1T:** Gate 3 Data Source 1 True Enable bit  
1 = The Data Source 1 signal is enabled for Gate 3  
0 = The Data Source 1 signal is disabled for Gate 3
- bit 16    **G3D1N:** Gate 3 Data Source 1 Negated Enable bit  
1 = The Data Source 1 inverted signal is enabled for Gate 3  
0 = The Data Source 1 inverted signal is disabled for Gate 3
- bit 15    **G2D4T:** Gate 2 Data Source 4 True Enable bit  
1 = The Data Source 4 signal is enabled for Gate 2  
0 = The Data Source 4 signal is disabled for Gate 2
- bit 14    **G2D4N:** Gate 2 Data Source 4 Negated Enable bit  
1 = The Data Source 4 inverted signal is enabled for Gate 2  
0 = The Data Source 4 inverted signal is disabled for Gate 2
- bit 13    **G2D3T:** Gate 2 Data Source 3 True Enable bit  
1 = The Data Source 3 signal is enabled for Gate 2  
0 = The Data Source 3 signal is disabled for Gate 2
- bit 12    **G2D3N:** Gate 2 Data Source 3 Negated Enable bit  
1 = The Data Source 3 inverted signal is enabled for Gate 2  
0 = The Data Source 3 inverted signal is disabled for Gate 2
- bit 11    **G2D2T:** Gate 2 Data Source 2 True Enable bit  
1 = The Data Source 2 signal is enabled for Gate 2  
0 = The Data Source 2 signal is disabled for Gate 2
- bit 10    **G2D2N:** Gate 2 Data Source 2 Negated Enable bit  
1 = The Data Source 2 inverted signal is enabled for Gate 2  
0 = The Data Source 2 inverted signal is disabled for Gate 2
- bit 9     **G2D1T:** Gate 2 Data Source 1 True Enable bit  
1 = The Data Source 1 signal is enabled for Gate 2  
0 = The Data Source 1 signal is disabled for Gate 2
- bit 8     **G2D1N:** Gate 2 Data Source 1 Negated Enable bit  
1 = The Data Source 1 inverted signal is enabled for Gate 2  
0 = The Data Source 1 inverted signal is disabled for Gate 2
- bit 7     **G1D4T:** Gate 1 Data Source 4 True Enable bit  
1 = The Data Source 4 signal is enabled for Gate 1  
0 = The Data Source 4 signal is disabled for Gate 1
- bit 6     **G1D4N:** Gate 1 Data Source 4 Negated Enable bit  
1 = The Data Source 4 inverted signal is enabled for Gate 1  
0 = The Data Source 4 inverted signal is disabled for Gate 1
- bit 5     **G1D3T:** Gate 1 Data Source 3 True Enable bit  
1 = The Data Source 3 signal is enabled for Gate 1  
0 = The Data Source 3 signal is disabled for Gate 1

---

---

**REGISTER 20-3: CLCxGLS: CLCx GATE LOGIC INPUT SELECT REGISTER (CONTINUED)**

- bit 4    **G1D3N:** Gate 1 Data Source 3 Negated Enable bit  
1 = The Data Source 3 inverted signal is enabled for Gate 1  
0 = The Data Source 3 inverted signal is disabled for Gate 1
- bit 3    **G1D2T:** Gate 1 Data Source 2 True Enable bit  
1 = The Data Source 2 signal is enabled for Gate 1  
0 = The Data Source 2 signal is disabled for Gate 1
- bit 2    **G1D2N:** Gate 1 Data Source 2 Negated Enable bit  
1 = The Data Source 2 inverted signal is enabled for Gate 1  
0 = The Data Source 2 inverted signal is disabled for Gate 1
- bit 1    **G1D1T:** Gate 1 Data Source 1 True Enable bit  
1 = The Data Source 1 signal is enabled for Gate 1  
0 = The Data Source 1 signal is disabled for Gate 1
- bit 0    **G1D1N:** Gate 1 Data Source 1 Negated Enable bit  
1 = The Data Source 1 inverted signal is enabled for Gate 1  
0 = The Data Source 1 inverted signal is disabled for Gate 1

# PIC32MK GPG/MCJ with CAN FD Family

## 22.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 29. “Real-Time Clock and Calendar (RTCC)”** (DS60001125), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

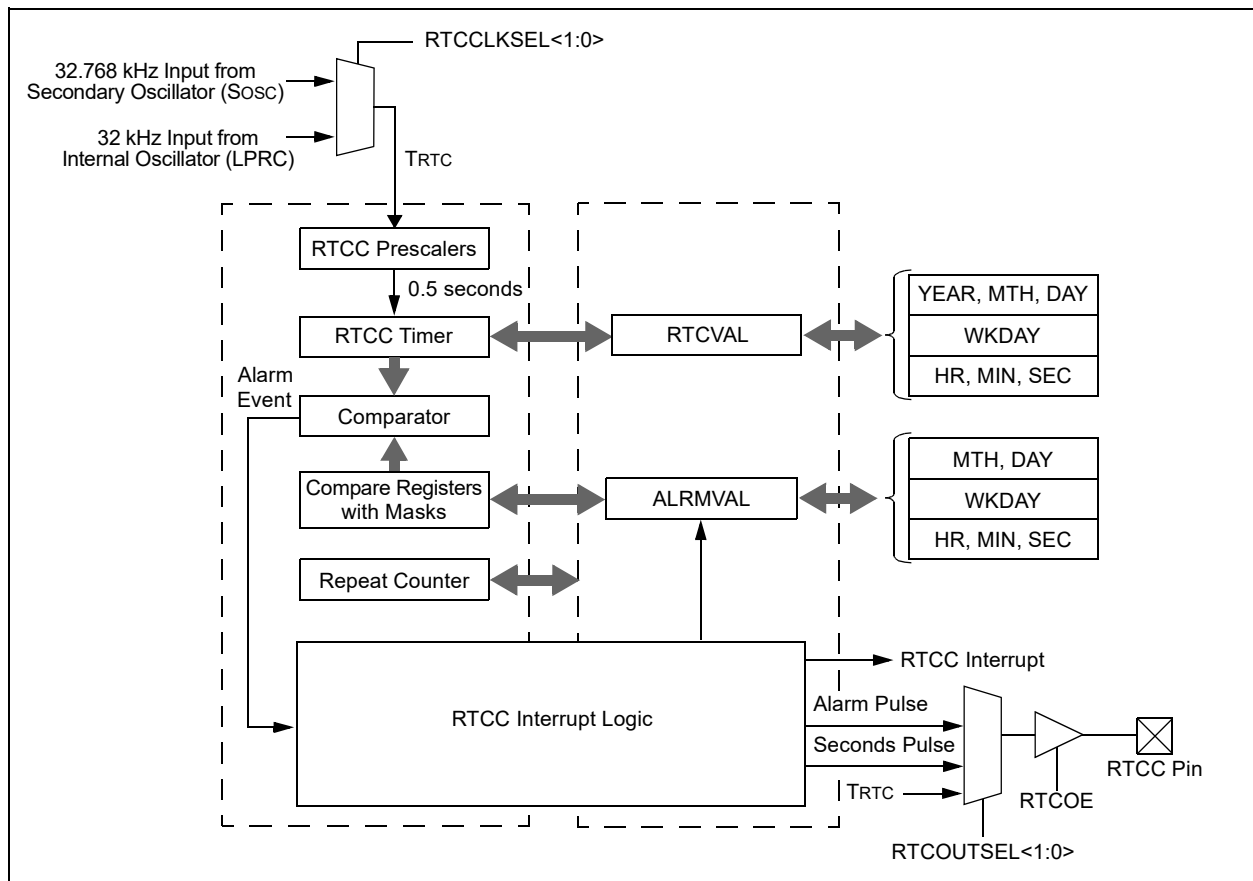
The RTCC module is intended for applications in which accurate time must be maintained for extended periods of time with minimal or no CPU intervention. Low-power optimization provides extended battery lifetime while keeping track of time.

The following are key features of the RTCC module:

- Time: hours, minutes and seconds

- 24-hour format (military time)
- Visibility of one-half second period
- Provides calendar: Weekday, date, month and year
- Alarm intervals are configurable for half of a second, one second, 10 seconds, one minute, 10 minutes, one hour, one day, one week, one month, and one year
- Alarm repeat with decremting counter
- Alarm with indefinite repeat: Chime
- Year range: 2000 to 2099
- Leap year correction
- BCD format for smaller firmware overhead
- Optimized for long-term battery operation
- Fractional second synchronization
- User calibration of the clock crystal frequency with auto-adjust
- Calibration range:  $\pm 0.66$  seconds error per month
- Calibrates up to 260 ppm of crystal error
- Uses external 32.768 kHz crystal or 32 kHz internal oscillator
- Alarm pulse, seconds clock, or internal clock output on RTCC pin

**FIGURE 22-1: RTCC BLOCK DIAGRAM**



## 22.1 RTCC Control Registers

**TABLE 22-1: RTCC REGISTER MAP**

| Virtual Address<br>(BF8C_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits        |       |       |          |             |                |                |          |              |           |         |         |              |       |      | All Resets |
|-----------------------------|---------------------------------|-----------|-------------|-------|-------|----------|-------------|----------------|----------------|----------|--------------|-----------|---------|---------|--------------|-------|------|------------|
|                             |                                 |           | 31/15       | 30/14 | 29/13 | 28/12    | 27/11       | 26/10          | 25/9           | 24/8     | 23/7         | 22/6      | 21/5    | 20/4    | 19/3         | 18/2  | 17/1 |            |
| 0000                        | RTCCON                          | 31:16     | —           | —     | —     | —        | —           | —              | CAL<9:0>       |          |              |           |         |         |              |       |      | 0000       |
|                             |                                 | 15:0      | ON          | —     | SIDL  | —        | —           | RTCCLKSEL<1:0> | RTCOUTSEL<1:0> | RTCCLKON | —            | —         | RTCWREN | RTCSYNC | HALFSEC      | RTCOC | 0000 |            |
| 0010                        | RTCALRM                         | 31:16     | —           | —     | —     | —        | —           | —              | —              | —        | —            | —         | —       | —       | —            | —     | —    | 0000       |
|                             |                                 | 15:0      | ALRMEN      | CHIME | PIV   | ALRMSYNC | AMASK<3:0>  |                |                |          |              | ARPT<7:0> |         |         |              |       | 0000 |            |
| 0020                        | RTCTIME                         | 31:16     | HR10<3:0>   |       |       |          | HR01<3:0>   |                |                |          | MIN10<3:0>   |           |         |         | MIN01<3:0>   |       |      | xxxx       |
|                             |                                 | 15:0      | SEC10<3:0>  |       |       |          | SEC01<3:0>  |                |                |          | —            | —         | —       | —       | —            | —     | —    | —          |
| 0030                        | RTCDATE                         | 31:16     | YEAR10<3:0> |       |       |          | YEAR01<3:0> |                |                |          | MONTH10<3:0> |           |         |         | MONTH01<3:0> |       |      | xxxx       |
|                             |                                 | 15:0      | DAY10<3:0>  |       |       |          | DAY01<3:0>  |                |                |          | —            | —         | —       | —       | WDAY01<3:0>  |       |      | xx00       |
| 0040                        | ALRMTIME                        | 31:16     | HR10<3:0>   |       |       |          | HR01<3:0>   |                |                |          | MIN10<3:0>   |           |         |         | MIN01<3:0>   |       |      | xxxx       |
|                             |                                 | 15:0      | SEC10<3:0>  |       |       |          | SEC01<3:0>  |                |                |          | —            | —         | —       | —       | —            | —     | —    | —          |
| 0050                        | ALRMDATE                        | 31:16     | —           | —     | —     | —        | —           | —              | —              | —        | MONTH10<3:0> |           |         |         | MONTH01<3:0> |       |      | 00xx       |
|                             |                                 | 15:0      | DAY10<3:0>  |       |       |          | DAY01<3:0>  |                |                |          | —            | —         | —       | —       | WDAY01<3:0>  |       |      | xx0x       |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 22-1: RTCCON: REAL-TIME CLOCK AND CALENDAR CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7               | Bit 30/22/14/6           | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3          | Bit 26/18/10/2 | Bit 25/17/9/1          | Bit 24/16/8/0                |
|-----------|------------------------------|--------------------------|----------------|----------------|-------------------------|----------------|------------------------|------------------------------|
| 31:24     | U-0                          | U-0                      | U-0            | U-0            | U-0                     | U-0            | R/W-0                  | R/W-0                        |
|           | —                            | —                        | —              | —              | —                       | —              | CAL<9:8>               |                              |
| 23:16     | R/W-0                        | R/W-0                    | R/W-0          | R/W-0          | R/W-0                   | R/W-0          | R/W-0                  | R/W-0                        |
|           | CAL<7:0>                     |                          |                |                |                         |                |                        |                              |
| 15:8      | R/W-0                        | U-0                      | R/W-0          | U-0            | U-0                     | R/W-0          | R/W-0                  | R/W-0                        |
|           | ON <sup>(1)</sup>            | —                        | SIDL           | —              | —                       | RTCCLKSEL<1:0> |                        | RTC OUTSEL<1> <sup>(2)</sup> |
| 7:0       | R/W-0                        | R-0                      | U-0            | U-0            | R/W-0                   | R-0            | R-0                    | R/W-0                        |
|           | RTC OUTSEL<0> <sup>(2)</sup> | RTC CLKON <sup>(5)</sup> | —              | —              | RTC WREN <sup>(3)</sup> | RTC SYNC       | HALFSEC <sup>(4)</sup> | RTC OE                       |

|                   |                  |                                    |                    |
|-------------------|------------------|------------------------------------|--------------------|
| <b>Legend:</b>    |                  |                                    |                    |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |                    |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               | x = Bit is unknown |

bit 31-26 **Unimplemented:** Read as '0'

bit 25-16 **CAL<9:0>:** Real-Time Clock Drift Calibration bits, which contain a signed 10-bit integer value

0111111111 = Maximum positive adjustment, adds 511 real-time clock pulses every one minute

•

•

•

0000000001 = Minimum positive adjustment, adds 1 real-time clock pulse every one minute

0000000000 = No adjustment

1111111111 = Minimum negative adjustment, subtracts 1 real-time clock pulse every one minute

•

•

1000000000 = Maximum negative adjustment, subtracts 512 real-time clock pulses every one minute

bit 15 **ON:** RTCC On bit<sup>(1)</sup>

1 = RTCC module is enabled

0 = RTCC module is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Stop in Idle Mode bit

1 = Disables RTCC operation when CPU enters Idle mode

0 = Continue normal operation when CPU enters Idle mode

bit 12-11 **Unimplemented:** Read as '0'

**Note 1:** The ON bit is only writable after RTCWREN = 1.

**2:** Requires RTCOE = 1 (RTCCON<0>) for the output to be active.

**3:** The RTCWREN bit can be set only when the write sequence is enabled.

**4:** This bit is read-only. It is cleared to '0' on a write to the seconds bit fields (RTCTIME<14:8>).

**5:** This bit is undefined when RTCCLKSEL<1:0> = 00 (LPRC is the clock source).

**Note:** This register is reset only on a POR.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 22-1: RTCCON: REAL-TIME CLOCK AND CALENDAR CONTROL REGISTER (CONTINUED)

bit 10-9 **RTCCLKSEL<1:0>**: RTCC Clock Select bits

When a new value is written to these bits, the Seconds Value register should also be written to properly reset the clock prescalers in the RTCC.

11 = Reserved

10 = Reserved

01 = RTCC uses the external 32.768 kHz Secondary Oscillator (SOSC)

00 = RTCC uses the internal 32 kHz oscillator (LPRC)

bit 8-7 **RTCOUTSEL<1:0>**: RTCC Output Data Select bits<sup>(2)</sup>

11 = Reserved

10 = RTCC Clock is presented on the RTCC pin

01 = Seconds Clock is presented on the RTCC pin

00 = Alarm Pulse is presented on the RTCC pin when the alarm interrupt is triggered

bit 6 **RTCCLKON**: RTCC Clock Enable Status bit<sup>(5)</sup>

1 = RTCC Clock is actively running

0 = RTCC Clock is not running

bit 5-4 **Unimplemented**: Read as '0'

bit 3 **RTCWREN**: Real-Time Clock Value Registers Write Enable bit<sup>(3)</sup>

1 = Real-Time Clock Value registers can be written to by the user

0 = Real-Time Clock Value registers are locked out from being written to by the user

bit 2 **RTCSYNC**: Real-Time Clock Value Registers Read Synchronization bit

1 = Real-time clock value registers can change while reading (due to a rollover ripple that results in an invalid data read). If the register is read twice and results in the same data, the data can be assumed to be valid.

0 = Real-time clock value registers can be read without concern about a rollover ripple

bit 1 **HALFSEC**: Half-Second Status bit<sup>(4)</sup>

1 = Second half period of a second

0 = First half period of a second

bit 0 **RTC OE**: RTCC Output Enable bit

1 = RTCC output is enabled

0 = RTCC output is not enabled

**Note 1:** The ON bit is only writable after RTCWREN = 1.

**2:** Requires RTCOE = 1 (RTCCON<0>) for the output to be active.

**3:** The RTCWREN bit can be set only when the write sequence is enabled.

**4:** This bit is read-only. It is cleared to '0' on a write to the seconds bit fields (RTCTIME<14:8>).

**5:** This bit is undefined when RTCCLKSEL<1:0> = 00 (LPRC is the clock source).

**Note:** This register is reset only on a POR.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 22-2: RTCALRM: REAL-TIME CLOCK ALARM CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7           | Bit 30/22/14/6       | Bit 29/21/13/5     | Bit 28/20/12/4 | Bit 27/19/11/3            | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|--------------------------|----------------------|--------------------|----------------|---------------------------|----------------|---------------|---------------|
| 31:24     | U-0                      | U-0                  | U-0                | U-0            | U-0                       | U-0            | U-0           | U-0           |
|           | —                        | —                    | —                  | —              | —                         | —              | —             | —             |
| 23:16     | U-0                      | U-0                  | U-0                | U-0            | U-0                       | U-0            | U-0           | U-0           |
|           | —                        | —                    | —                  | —              | —                         | —              | —             | —             |
| 15:8      | R/W-0                    | R/W-0                | R/W-0              | R-0            | R/W-0                     | R/W-0          | R/W-0         | R/W-0         |
|           | ALRMEN <sup>(1,2)</sup>  | CHIME <sup>(2)</sup> | PIV <sup>(2)</sup> | ALRMSYNC       | AMASK<3:0> <sup>(2)</sup> |                |               |               |
| 7:0       | R/W-0                    | R/W-0                | R/W-0              | R/W-0          | R/W-0                     | R/W-0          | R/W-0         | R/W-0         |
|           | ARPT<7:0> <sup>(2)</sup> |                      |                    |                |                           |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ALRMEN:** Alarm Enable bit<sup>(1,2)</sup>  
 1 = Alarm is enabled  
 0 = Alarm is disabled

bit 14 **CHIME:** Chime Enable bit<sup>(2)</sup>  
 1 = Chime is enabled – ARPT<7:0> is allowed to rollover from 0x00 to 0xFF  
 0 = Chime is disabled – ARPT<7:0> stops once it reaches 0x00

bit 13 **PIV:** Alarm Pulse Initial Value bit<sup>(2)</sup>  
 When ALRMEN = 0, PIV is writable and determines the initial value of the Alarm Pulse.  
 When ALRMEN = 1, PIV is read-only and returns the state of the Alarm Pulse.

bit 12 **ALRMSYNC:** Alarm Sync bit  
 1 = ARPT<7:0> and ALRMEN may change as a result of a half second rollover during a read.  
 The ARPT must be read repeatedly until the same value is read twice. This must be done since multiple bits may be changing.  
 0 = ARPT<7:0> and ALRMEN can be read without concerns of rollover because the prescaler is more than 32 real-time clocks away from a half-second rollover

bit 11-8 **AMASK<3:0>:** Alarm Mask Configuration bits<sup>(2)</sup>  
 0000 = Every half-second  
 0001 = Every second  
 0010 = Every 10 seconds  
 0011 = Every minute  
 0100 = Every 10 minutes  
 0101 = Every hour  
 0110 = Once a day  
 0111 = Once a week  
 1000 = Once a month  
 1001 = Once a year (except when configured for February 29, once every four years)  
 1010 = Reserved  
 1011 = Reserved  
 11xx = Reserved

- Note 1:** Hardware clears the ALRMEN bit anytime the alarm event occurs, when ARPT<7:0> = 00 and CHIME = 0.  
**2:** This field should not be written when the RTCC ON bit = '1' (RTCCON<15>) and ALRMSYNC = 1.

**Note:** The RTCALRM register is reset on a MCLR or Power-on Reset (POR).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 22-2: RTCALRM: REAL-TIME CLOCK ALARM CONTROL REGISTER (CONTINUED)

bit 7-0 **ARPT<7:0>**: Alarm Repeat Counter Value bits<sup>(2)</sup>

11111111 =Alarm will trigger 256 times

.

.

00000000 =Alarm will trigger one time

The counter decrements on any alarm event. The counter only rolls over from 0x00 to 0xFF if CHIME = 1.

**Note 1:** Hardware clears the ALRMEN bit anytime the alarm event occurs, when ARPT<7:0> = 00 and CHIME = 0.

**2:** This field should not be written when the RTCC ON bit = '1' (RTCCON<15>) and ALRMSYNC = 1.

|   |
|---|
| <b>Note:</b> The RTCALRM register is reset on a MCLR or Power-on Reset (POR). |
|---|

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 22-3: RTCTIME: REAL-TIME CLOCK TIME VALUE REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | HR10<3:0>      |                |                |                | HR01<3:0>      |                |               |               |
| 23:16     | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | MIN10<3:0>     |                |                |                | MIN01<3:0>     |                |               |               |
| 15:8      | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | SEC10<3:0>     |                |                |                | SEC01<3:0>     |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 31-28 **HR10<3:0>**: Binary-Coded Decimal Value of Hours bits, 10 digits; contains a value from 0 to 2
- bit 27-24 **HR01<3:0>**: Binary-Coded Decimal Value of Hours bits, 1 digit; contains a value from 0 to 9
- bit 23-20 **MIN10<3:0>**: Binary-Coded Decimal Value of Minutes bits, 10 digits; contains a value from 0 to 5
- bit 19-16 **MIN01<3:0>**: Binary-Coded Decimal Value of Minutes bits, 1 digit; contains a value from 0 to 9
- bit 15-12 **SEC10<3:0>**: Binary-Coded Decimal Value of Seconds bits, 10 digits; contains a value from 0 to 5
- bit 11-8 **SEC01<3:0>**: Binary-Coded Decimal Value of Seconds bits, 1 digit; contains a value from 0 to 9
- bit 7-0 **Unimplemented**: Read as '0'

**Note:** This register is only writable when RTCWREN = 1 (RTCCON<3>).



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 22-5: ALRMTIME: ALARM TIME VALUE REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | HR10<3:0>      |                |                |                | HR01<3:0>      |                |               |               |
| 23:16     | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | MIN10<3:0>     |                |                |                | MIN01<3:0>     |                |               |               |
| 15:8      | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | SEC10<3:0>     |                |                |                | SEC01<3:0>     |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-28 **HR10<3:0>**: Binary Coded Decimal value of hours bits, 10 digits; contains a value from 0 to 2

bit 27-24 **HR01<3:0>**: Binary Coded Decimal value of hours bits, 1 digit; contains a value from 0 to 9

bit 23-20 **MIN10<3:0>**: Binary Coded Decimal value of minutes bits, 10 digits; contains a value from 0 to 5

bit 19-16 **MIN01<3:0>**: Binary Coded Decimal value of minutes bits, 1 digit; contains a value from 0 to 9

bit 15-12 **SEC10<3:0>**: Binary Coded Decimal value of seconds bits, 10 digits; contains a value from 0 to 5

bit 11-8 **SEC01<3:0>**: Binary Coded Decimal value of seconds bits, 1 digit; contains a value from 0 to 9

bit 7-0 **Unimplemented**: Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 22-6: ALRMDATE: ALARM DATE VALUE REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—      | U-0<br>—      |
| 23:16     | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | MONTH10<3:0>   |                |                |                | MONTH01<3:0>   |                |               |               |
| 15:8      | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | DAY10<1:0>     |                |                |                | DAY01<3:0>     |                |               |               |
| 7:0       | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—       | R/W-x          | R/W-x          | R/W-x         | R/W-x         |
|           | WDAY01<3:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31-24 **Unimplemented:** Read as '0'
- bit 23-20 **MONTH10<3:0>**: Binary Coded Decimal value of months bits, 10 digits; contains a value from 0 to 1
- bit 19-16 **MONTH01<3:0>**: Binary Coded Decimal value of months bits, 1 digit; contains a value from 0 to 9
- bit 15-12 **DAY10<3:0>**: Binary Coded Decimal value of days bits, 10 digits; contains a value from 0 to 3
- bit 11-8 **DAY01<3:0>**: Binary Coded Decimal value of days bits, 1 digit; contains a value from 0 to 9
- bit 7-4 **Unimplemented:** Read as '0'
- bit 3-0 **WDAY01<3:0>**: Binary Coded Decimal value of weekdays bits, 1 digit; contains a value from 0 to 6

# PIC32MK GPG/MCJ with CAN FD Family

## 23.0 12-BIT HIGH-SPEED SUCCESSIVE APPROXIMATION REGISTER (SAR) ANALOG-TO-DIGITAL CONVERTER (ADC)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 22. “12-bit High-Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC)”** (DS60001344) in the “PIC32 Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

The 12-bit High-Speed Successive Approximation Register (SAR) analog-to-digital converter (ADC) includes the following features:

- 12-bit resolution
- Seven ADC modules with dedicated Sample and Hold (S&H) circuits
- Up to 6 dedicated class1 ADC modules can be combined using an interleave technique to provide conversion rates up to 20 msp/s, refer to Application Note: “AN2785\_World's Fastest Embedded Interleaved 12-bit ADC Using PIC32MZ and PIC32MK Families”.
- Up to 45 analog input sources, in addition to the internal CTMU, internal voltage reference and internal temperature sensor
- Single-ended and/or differential inputs
- Supports touch sense applications
- Four digital comparators
- Four digital filters supporting two modes:
  - Oversampling mode
  - Averaging mode
- Early interrupt generation resulting in faster processing of converted data
- Designed for power conversion and general purpose applications
- Operation during Sleep and Idle modes

A simplified block diagram of the ADC module is illustrated in [Figure 23-1](#).

The 12-bit HS SAR ADC has up to six dedicated ADC modules (ADC0-ADC5) and one shared ADC module (ADC7). The dedicated ADC modules use a single input (or its alternate) and are intended for high-speed and precise sampling of time-sensitive or transient inputs. The shared ADC module incorporates a multiplexer on the input to facilitate a larger group of inputs, with slower sampling, and provides flexible automated scanning option through the input scan logic.

For each ADC module, the analog inputs are connected to the S&H capacitor. The clock, sampling time, and output data resolution for each ADC module can be set independently. The ADC module performs the conversion of the input analog signal based on the configurations set in the registers. When conversion is complete, the final result is stored in the result buffer for the specific analog input and is passed to the digital filter and digital comparator if configured to use data from this particular sample. Input to ADCx mapping is illustrated in [Figure 23-2](#).

### 23.1 Activation Sequence

The following ADCx activation sequence is to be followed at all times:

**Step 1:** Initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

Then, configure the AICMPEN bit (ADCCON1<12> and the IOANCPEN bit (CFGCON<7>) = 1 if and only if VDD is less than 2.5V. The default is '0', which assumes VDD is greater than or equal to 2.5V.

**Step 2:** The user writes all the essential ADC configuration SFRs including the ADC control clock and all ADC core clocks setup:

- ADCCON1, keeping the ON bit = 0
- ADCCON2, especially paying attention to ADCDIV<6:0> and SAMC<9:0>
- ADCANCON, keeping all analog enables ANENx bit = 0, WKUPCLKCNT bit = 0xA
- ADCCON3, keeping all DIGEN5x = 0, especially paying attention to ADCSEL<1:0>, CONCLKDIV <5:0>, and VREFSEL<2:0>
- ADCxTIME, ADCDIVx<6:0>, and SAMCx<9:0>
- ADCTRGMODE, ADCIMCONx, ADCTRGSNS, ADCCSSx, ADCGIRQENx, ADCTRGx, ADCBASE
- Comparators, Filters, etc.

**Step 3:** The user sets the ANENx bit to '1' for the ADC SAR Cores needed (which internally in the ADC module enables the control clock to generate by division the core clocks for the desired ADC SAR Cores, which in turn enables the bias circuitry for these ADC SAR Cores).

**Step 4:** The user sets the ON bit to '1', which enables the ADC control clock.

**Step 5:** The user waits for the interrupt/polls the BGVRDY bit (ADCCON2<31>) and the WKRDYx bit (ADCANCON<15,13:8>) = 1, which signals that the device analog environment (band gap and VREF) is ready.

# PIC32MK GPG/MCJ with CAN FD Family

**Step 6:** Set the DIGENx bit (ADCCON3<15,13:8>) to '1', which enables the digital circuitry to immediately begin processing incoming triggers to perform data conversions.

**Note:** For the best optimized CPU and ISR performance the user should refer to [Table 6-1](#). The CPU interrupt latency is ~43 SYSCLK cycles if no other interrupts are pending. If not using ADC DMA and the ADC combined sum throughput rate of all the ADC modules in use is greater than  $(SYSCLK / 43) = 2.8$  Msps, it is recommended to use the ADC CPU early interrupt generation, defined in the ADCxTIME and ADCEIENx registers. This will reduce the probability of the ADC results being overwritten by the next conversion before the CPU can read the previous ADC result(s).  
Do not use the early interrupts if using the ADC in the DMA module.

Do not activate ADC trigger sources until ADC has been completely initialized, enabled, and warm up time complete.

**Note:** If using ADC DMA, ADC source clock must be SYSCLK only.

Non-interleaved Dedicated Class\_1 ADCx Throughput rate

$$= 1 / ((\text{Sample time} + \text{Conversion time})(TAD))$$

$$= 1 / ((S\text{AMC} + \# \text{ bit resolution} + 1)(TAD))$$

For example:

SAMC = 3 TAD, 12-bit mode, TAD = 16.667 ns = 60 MHz

Throughput rate:

$$= 1 / ((3 + 12 + 1)(16.667 \text{ ns}))$$

$$= 1 / (16 * 16.667 \text{ ns})$$

$$= 3.75 \text{ msp}$$

**TABLE 23-1: PIC32MKXXX BASED ON 60 MHZ TAD CLOCK (16.667 ns)**

| Number of Interleaved ADCs Used | 12-bit (Max) Msps | 10-bit (Max) Msps | 8-bit (Max) Msps | 6-bit (Max) Msps |
|---------------------------------|-------------------|-------------------|------------------|------------------|
| 1                               | 3.75              | 4.286             | 5.0              | 6.0              |
| 2                               | 7.50              | 8.571             | 10.00            | 12.00            |
| 3                               | 10.00             | 12.00             | 15.00            | 15.00            |
| 4                               | 15.00             | 17.1429           | 20.00            | 24.00            |
| 5                               | 15.00             | 20.00             | 20.00            | 30.00            |
| 6 <sup>(2)</sup>                | 20.00             | 24.00             | 30.00            | 30.00            |

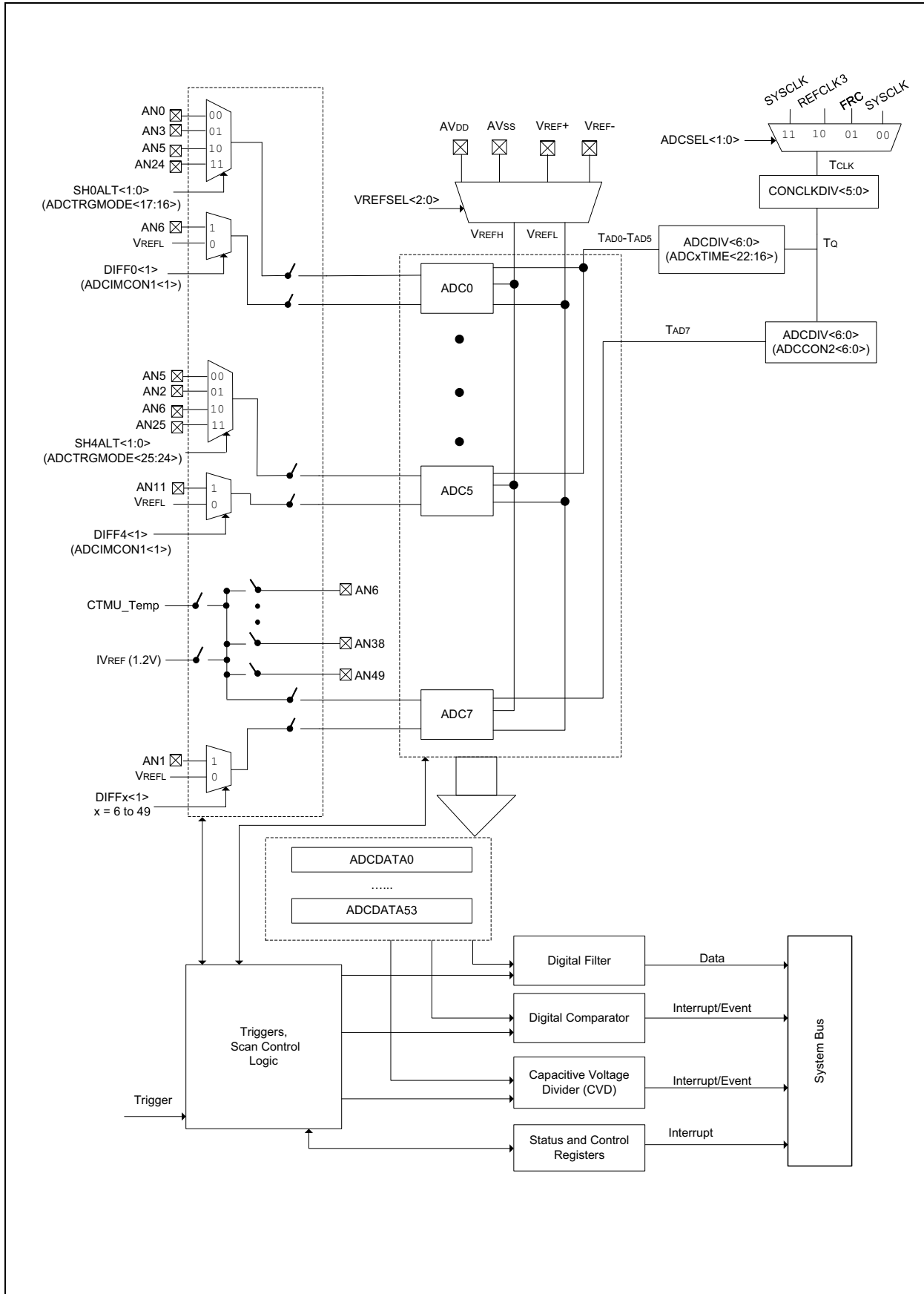
**Note 1:** Interleaved ADCs in this context means connecting the same analog source signal to multiple dedicated Class\_1 ADCs (that is, ADC0-ADC5), and using independent staggered trigger sources accordingly for each interleaved ADC.

**2:** Only available in the Motor Control Variant, that is, PIC32MKXXMCXX.



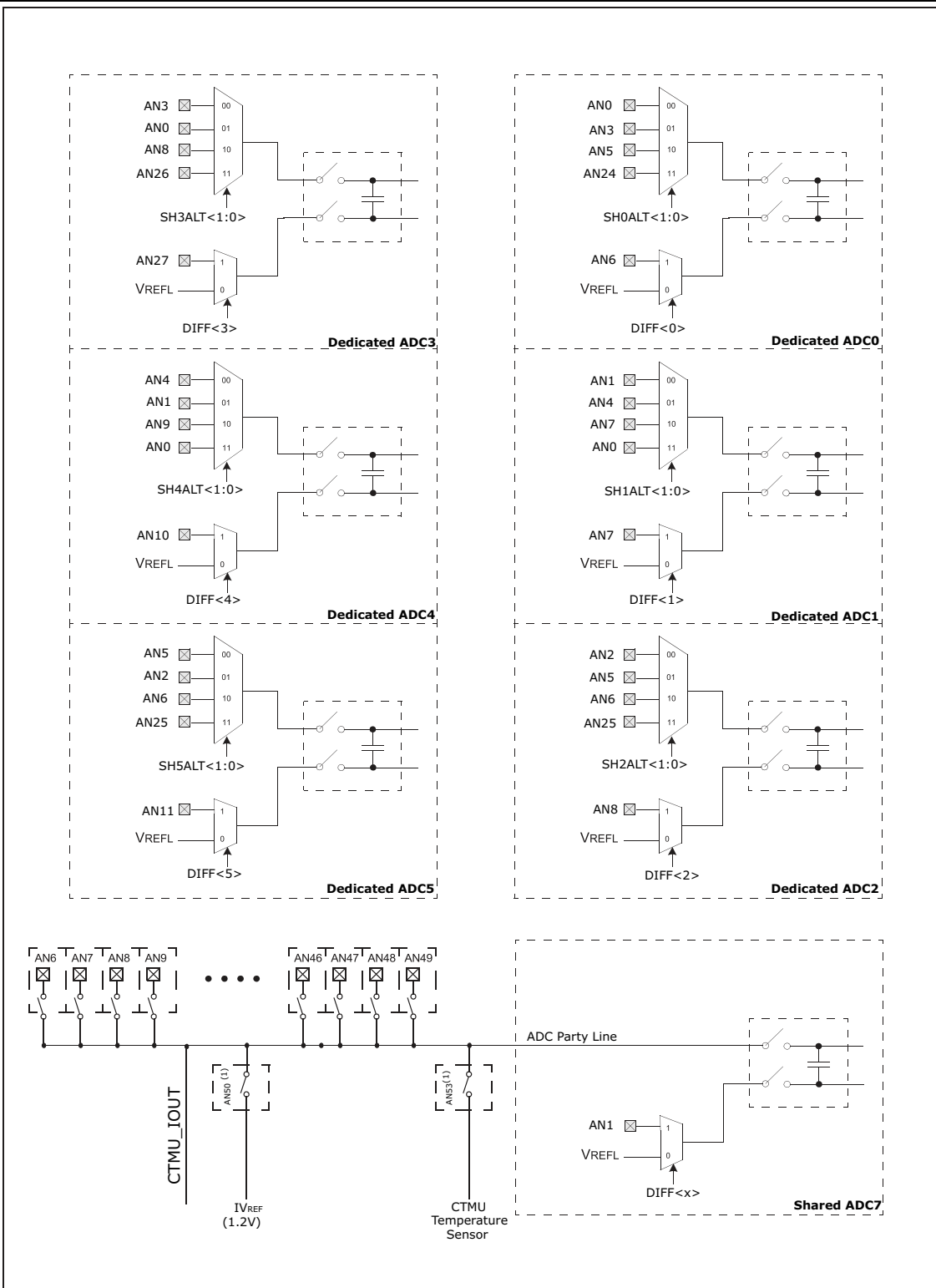
# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 23-1: ADC BLOCK DIAGRAM



# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 23-2: S&H BLOCK DIAGRAM**



**Note 1:** AN50 and AN53 are internal analog input sources.



TABLE 23-2: ADC REGISTER MAP (CONTINUED)

| Virtual Address | Register Name          | Bit Range | Bits           |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | All Resets |      |
|-----------------|------------------------|-----------|----------------|----------|--------|------------------------------|--------|--------|--------|--------|-----------------------|-----------------------|-----------------------|-----------------------|--------|--------|--------|------------|------|
|                 |                        |           | 31/15          | 30/14    | 29/13  | 28/12                        | 27/11  | 26/10  | 25/9   | 24/8   | 23/7                  | 22/6                  | 21/5                  | 20/4                  | 19/3   | 18/2   | 17/1   |            | 16/0 |
| 7100            | ADCCMPEN2              | 31:16     | —              | —        | —      | —                            | CMPE27 | CMPE26 | CMPE25 | CMPE24 | CMPE23 <sup>(1)</sup> | CMPE22 <sup>(1)</sup> | CMPE21 <sup>(1)</sup> | CMPE20 <sup>(1)</sup> | CMPE19 | CMPE18 | CMPE17 | CMPE16     | 0000 |
|                 |                        | 15:0      | CMPE15         | CMPE14   | CMPE13 | CMPE12                       | CMPE11 | CMPE10 | CMPE9  | CMPE8  | CMPE7                 | CMPE6                 | CMPE5                 | CMPE4                 | CMPE3  | CMPE2  | CMPE1  | CMPE0      | 0000 |
| 7110            | ADCCMP2                | 31:16     | DCMPHI<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
|                 |                        | 15:0      | DCMPLO<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7120            | ADCCMPEN3              | 31:16     | —              | —        | —      | —                            | CMPE27 | CMPE26 | CMPE25 | CMPE24 | CMPE23 <sup>(1)</sup> | CMPE22 <sup>(1)</sup> | CMPE21 <sup>(1)</sup> | CMPE20 <sup>(1)</sup> | CMPE19 | CMPE18 | CMPE17 | CMPE16     | 0000 |
|                 |                        | 15:0      | CMPE15         | CMPE14   | CMPE13 | CMPE12                       | CMPE11 | CMPE10 | CMPE9  | CMPE8  | CMPE7                 | CMPE6                 | CMPE5                 | CMPE4                 | CMPE3  | CMPE2  | CMPE1  | CMPE0      | 0000 |
| 7130            | ADCCMP3                | 31:16     | DCMPHI<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
|                 |                        | 15:0      | DCMPLO<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7140            | ADCCMPEN4              | 31:16     | —              | —        | —      | —                            | CMPE27 | CMPE26 | CMPE25 | CMPE24 | CMPE23 <sup>(1)</sup> | CMPE22 <sup>(1)</sup> | CMPE21 <sup>(1)</sup> | CMPE20 <sup>(1)</sup> | CMPE19 | CMPE18 | CMPE17 | CMPE16     | 0000 |
|                 |                        | 15:0      | CMPE15         | CMPE14   | CMPE13 | CMPE12                       | CMPE11 | CMPE10 | CMPE9  | CMPE8  | CMPE7                 | CMPE6                 | CMPE5                 | CMPE4                 | CMPE3  | CMPE2  | CMPE1  | CMPE0      | 0000 |
| 7150            | ADCCMP4                | 31:16     | DCMPHI<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
|                 |                        | 15:0      | DCMPLO<15:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 71A0            | ADCFLTR1               | 31:16     | AFEN           | DATA16EN | DFMODE | OVRSAM<2:0>                  |        |        | AFGIEN | AFRDY  | —                     | —                     | —                     | CHNLID<4:0>           |        |        |        | 0000       |      |
|                 |                        | 15:0      | FLTRDATA<15:0> |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 71B0            | ADCFLTR2               | 31:16     | AFEN           | DATA16EN | DFMODE | OVRSAM<2:0>                  |        |        | AFGIEN | AFRDY  | —                     | —                     | —                     | CHNLID<4:0>           |        |        |        | 0000       |      |
|                 |                        | 15:0      | FLTRDATA<15:0> |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 71C0            | ADCFLTR3               | 31:16     | AFEN           | DATA16EN | DFMODE | OVRSAM<2:0>                  |        |        | AFGIEN | AFRDY  | —                     | —                     | —                     | CHNLID<4:0>           |        |        |        | 0000       |      |
|                 |                        | 15:0      | FLTRDATA<15:0> |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 71D0            | ADCFLTR4               | 31:16     | AFEN           | DATA16EN | DFMODE | OVRSAM<2:0>                  |        |        | AFGIEN | AFRDY  | —                     | —                     | —                     | CHNLID<4:0>           |        |        |        | 0000       |      |
|                 |                        | 15:0      | FLTRDATA<15:0> |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7200            | ADCTRG1                | 31:16     | —              | —        | —      | TRGSRC3<4:0>                 |        |        |        | —      | —                     | —                     | TRGSRC2<4:0>          |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC1<4:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7210            | ADCTRG2                | 31:16     | —              | —        | —      | TRGSRC7<4:0>                 |        |        |        | —      | —                     | —                     | TRGSRC6<4:0>          |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC5<4:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7220            | ADCTRG3                | 31:16     | —              | —        | —      | TRGSRC11<4:0>                |        |        |        | —      | —                     | —                     | TRGSRC10<4:0>         |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC9<4:0>   |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7230            | ADCTRG4                | 31:16     | —              | —        | —      | TRGSRC15<4:0>                |        |        |        | —      | —                     | —                     | TRGSRC14<4:0>         |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC13<4:0>  |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7240            | ADCTRG5                | 31:16     | —              | —        | —      | TRGSRC19<4:0> <sup>(1)</sup> |        |        |        | —      | —                     | —                     | TRGSRC18<4:0>         |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC17<4:0>  |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7250            | ADCTRG6 <sup>(1)</sup> | 31:16     | —              | —        | —      | TRGSRC23<4:0>                |        |        |        | —      | —                     | —                     | TRGSRC22<4:0>         |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC21<4:0>  |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |
| 7260            | ADCTRG7                | 31:16     | —              | —        | —      | TRGSRC27<4:0>                |        |        |        | —      | —                     | —                     | TRGSRC26<4:0>         |                       |        |        | 0000   |            |      |
|                 |                        | 15:0      | TRGSRC25<4:0>  |          |        |                              |        |        |        |        |                       |                       |                       |                       |        |        |        | 0000       |      |

**Note**

- 1: This bit or register is not available on 64-pin devices.
- 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
- 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

TABLE 23-2: ADC REGISTER MAP (CONTINUED)

| Virtual Address | Register Name | Bit Range | Bits                   |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | All Resets            |         |      |   |
|-----------------|---------------|-----------|------------------------|------------------------|------------------------|-------------|---------|---------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|---------|------|---|
|                 |               |           | 31/15                  | 30/14                  | 29/13                  | 28/12       | 27/11   | 26/10   | 25/9                  | 24/8                  | 23/7                  | 22/6                  | 21/5                  | 20/4                  | 19/3                  | 18/2                  |                       | 17/1    | 16/0 |   |
| 7280            | ADCCMPCON1    | 31:16     | CVDDATA<15:0>          |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
|                 |               | 15:0      | —                      | —                      | AINID<5:0>             |             |         |         | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
| 7290            | ADCCMPCON2    | 31:16     | —                      | —                      | —                      | —           | —       | —       | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
|                 |               | 15:0      | —                      | —                      | AINID<4:0>             |             |         |         | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
| 72A0            | ADCCMPCON3    | 31:16     | —                      | —                      | —                      | —           | —       | —       | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
|                 |               | 15:0      | —                      | —                      | AINID<4:0>             |             |         |         | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
| 72B0            | ADCCMPCON4    | 31:16     | —                      | —                      | —                      | —           | —       | —       | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
|                 |               | 15:0      | —                      | —                      | AINID<4:0>             |             |         |         | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
| 7300            | ADCBASE       | 31:16     | —                      | —                      | —                      | —           | —       | —       | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —       | —    | — |
|                 |               | 15:0      | ADCBASE<15:0>          |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
| 7310            | ADCDSTAT      | 31:16     | DMAEN                  | —                      | RBFIE5                 | RBFIE4      | RBFIE3  | RBFIE2  | RBFIE1                | RBFIE0                | WOVERR                | —                     | RBF5                  | RBF4                  | RBF3                  | RBF2                  | RBF1                  | RBF0    | 0000 |   |
|                 |               | 15:0      | DMACEN                 | —                      | RAFIEN5                | RAFIEN4     | RAFIEN3 | RAFIEN2 | RAFIEN1               | RAFIEN0               | —                     | —                     | RAF5                  | RAF4                  | RAF3                  | RAF2                  | RAF1                  | RAF0    | 0000 |   |
| 7320            | ADCCNTB       | 31:16     | ADCCNTB<31:16>         |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
|                 |               | 15:0      | ADCCNTB<15:0>          |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
| 7330            | ADCDMAB       | 31:16     | ADCDMAB<31:16>         |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
|                 |               | 15:0      | ADCDMAB<15:0>          |                        |                        |             |         |         |                       |                       |                       |                       |                       |                       |                       |                       | 0000                  |         |      |   |
| 7340            | ADCTRGNSNS    | 31:16     | —                      | —                      | —                      | —           | LVL27   | LVL26   | LVL25                 | LVL24                 | LVL23 <sup>(1)</sup>  | LVL22 <sup>(1)</sup>  | LVL21 <sup>(1)</sup>  | LVL20 <sup>(1)</sup>  | LVL19                 | LVL18                 | LVL17                 | LVL16   | 0000 |   |
|                 |               | 15:0      | LVL15                  | LVL14                  | LVL13                  | LVL12       | LVL11   | LVL10   | LVL9                  | LVL8                  | LVL7                  | LVL6                  | LVL5                  | LVL4                  | LVL3                  | LVL2                  | LVL1                  | LVL0    | 0000 |   |
| 7350            | ADC0TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 7360            | ADC1TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 7370            | ADC2TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 7380            | ADC3TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 7390            | ADC4TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 73A0            | ADC5TIME      | 31:16     | —                      | —                      | —                      | ADCEIS<2:0> |         |         | SELRES<1:0>           |                       | BCHEN                 | ADCDIV<6:0>           |                       |                       |                       |                       |                       | 0300    |      |   |
|                 |               | 15:0      | —                      | —                      | —                      | —           | —       | —       | SAMC<9:0>             |                       |                       |                       |                       |                       | 0000                  |                       |                       |         |      |   |
| 73C0            | ADCEIEN1      | 31:16     | —                      | —                      | —                      | —           | EIEN27  | EIEN26  | EIEN25                | EIEN24                | EIEN23 <sup>(1)</sup> | EIEN22 <sup>(1)</sup> | EIEN21 <sup>(1)</sup> | EIEN20 <sup>(1)</sup> | EIEN19                | EIEN18                | EIEN17                | EIEN16  | 0000 |   |
|                 |               | 15:0      | EIEN15                 | EIEN14                 | EIEN13                 | EIEN12      | EIEN11  | EIEN10  | EIEN9                 | EIEN8                 | EIEN7                 | EIEN6                 | EIEN5                 | EIEN4                 | EIEN3                 | EIEN2                 | EIEN1                 | EIEN0   | 0000 |   |
| 73D0            | ADCEIEN2      | 31:16     | —                      | —                      | —                      | —           | —       | —       | —                     | —                     | —                     | —                     | EIRDY53               | EIRDY52               | —                     | EIRDY50               | EIRDY49               | EIRDY48 | 0000 |   |
|                 |               | 15:0      | EIRDY47 <sup>(1)</sup> | EIRDY46 <sup>(1)</sup> | EIRDY45 <sup>(1)</sup> | —           | —       | —       | EIEN41 <sup>(1)</sup> | EIEN40 <sup>(1)</sup> | EIEN39 <sup>(1)</sup> | EIEN38 <sup>(1)</sup> | EIEN37 <sup>(1)</sup> | EIEN36 <sup>(1)</sup> | EIEN35 <sup>(1)</sup> | EIEN34 <sup>(1)</sup> | EIEN33 <sup>(1)</sup> | —       | 0000 |   |

**Note**

- 1: This bit or register is not available on 64-pin devices.
- 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
- 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

**TABLE 23-2: ADC REGISTER MAP (CONTINUED)**

| Virtual Address | Register Name | Bit Range | Bits                   |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | All Resets |
|-----------------|---------------|-----------|------------------------|------------------------|------------------------|---------|-----------------|---------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|---------|------------|
|                 |               |           | 31/15                  | 30/14                  | 29/13                  | 28/12   | 27/11           | 26/10   | 25/9                   | 24/8                   | 23/7                   | 22/6                   | 21/5                   | 20/4                   | 19/3                   | 18/2                   | 17/1                   | 16/0    |            |
| 73E0            | ADCEISTAT1    | 31:16     | —                      | —                      | —                      | —       | EIRDY27         | EIRDY26 | EIRDY25                | EIRDY24                | EIRDY23 <sup>(1)</sup> | EIRDY22 <sup>(1)</sup> | EIRDY21 <sup>(1)</sup> | EIRDY20 <sup>(1)</sup> | EIRDY19                | EIRDY18                | EIRDY17                | EIRDY16 | 0000       |
|                 |               | 15:0      | EIRDY15                | EIRDY14                | EIRDY13                | EIRDY12 | EIRDY11         | EIRDY10 | EIRDY9                 | EIRDY8                 | EIRDY7                 | EIRDY6                 | EIRDY5                 | EIRDY4                 | EIRDY3                 | EIRDY2                 | EIRDY1                 | EIRDY0  | 0000       |
| 73F0            | ADCEISTAT2    | 31:16     | —                      | —                      | —                      | —       | —               | —       | —                      | —                      | —                      | —                      | EIRDY53                | EIRDY52                | —                      | EIRDY50                | EIRDY49                | EIRDY48 | 0000       |
|                 |               | 15:0      | EIRDY47 <sup>(1)</sup> | EIRDY46 <sup>(1)</sup> | EIRDY45 <sup>(1)</sup> | —       | —               | —       | EIRDY41 <sup>(1)</sup> | EIRDY40 <sup>(1)</sup> | EIRDY39 <sup>(1)</sup> | EIRDY38 <sup>(1)</sup> | EIRDY37 <sup>(1)</sup> | EIRDY36 <sup>(1)</sup> | EIRDY35 <sup>(1)</sup> | EIRDY34 <sup>(1)</sup> | EIRDY33 <sup>(1)</sup> | —       | 0000       |
| 7400            | ADCANCON      | 31:16     | —                      | —                      | —                      | —       | WKUPCLKCNT<3:0> |         |                        |                        | WKIEN7                 | —                      | WKIEN5                 | WKIEN4                 | WKIEN3                 | WKIEN2                 | WKIEN1                 | WKIEN0  | 0000       |
|                 |               | 15:0      | WKRDY7                 | —                      | WKRDY5                 | WKRDY4  | WKRDY3          | WKRDY2  | WKRDY1                 | WKRDY0                 | ANEN7                  | —                      | ANEN5                  | ANEN4                  | ANEN3                  | ANEN2                  | ANEN1                  | ANEN0   | 0000       |
| 7600            | ADCDATA0      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7610            | ADCDATA1      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7620            | ADCDATA2      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7630            | ADCDATA3      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7640            | ADCDATA4      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7650            | ADCDATA5      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7660            | ADCDATA6      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7670            | ADCDATA7      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7680            | ADCDATA8      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 7690            | ADCDATA9      | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 76A0            | ADCDATA10     | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 76B0            | ADCDATA11     | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 76C0            | ADCDATA12     | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
| 76D0            | ADCDATA13     | 31:16     | DATA<31:16>            |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |
|                 |               | 15:0      | DATA<15:0>             |                        |                        |         |                 |         |                        |                        |                        |                        |                        |                        |                        |                        |                        |         | 0000       |

**Note**

- 1: This bit or register is not available on 64-pin devices.
- 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
- 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

**TABLE 23-2: ADC REGISTER MAP (CONTINUED)**

| Virtual Address | Register Name            | Bit Range | Bits        |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |
|-----------------|--------------------------|-----------|-------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|
|                 |                          |           | 31/15       | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            |
| 76E0            | ADCDATA14                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 76F0            | ADCDATA15                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7700            | ADCDATA16                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7710            | ADCDATA17                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7720            | ADCDATA18                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7730            | ADCDATA19                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7740            | ADCDATA20 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7750            | ADCDATA21 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7760            | ADCDATA22 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7770            | ADCDATA23 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7780            | ADCDATA24                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7790            | ADCDATA25                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 77A0            | ADCDATA26                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 77B0            | ADCDATA27                | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7810            | ADCDATA33 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7820            | ADCDATA34 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7830            | ADCDATA35 <sup>(1)</sup> | 31:16     | DATA<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                          | 15:0      | DATA<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |

**Note**

- 1: This bit or register is not available on 64-pin devices.
- 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
- 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

TABLE 23-2: ADC REGISTER MAP (CONTINUED)

| Virtual Address | Register Name            | Bit Range | Bits                |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | All Resets |      |
|-----------------|--------------------------|-----------|---------------------|---------------------|---------------------|-------|-------|-------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|------------|------|
|                 |                          |           | 31/15               | 30/14               | 29/13               | 28/12 | 27/11 | 26/10 | 25/9                | 24/8                | 23/7                | 22/6                | 21/5                | 20/4                | 19/3                | 18/2                | 17/1                |            | 16/0 |
| 7840            | ADCDATA36 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7850            | ADCDATA37 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7860            | ADCDATA38 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7870            | ADCDATA39 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7880            | ADCDATA40 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7890            | ADCDATA41 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 78D0            | ADCDATA45 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 78E0            | ADCDATA46 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 78F0            | ADCDATA47 <sup>(1)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7900            | ADCDATA48                | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7910            | ADCDATA49                | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7920            | ADCDATA50 <sup>(2)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7950            | ADCDATA53 <sup>(2)</sup> | 31:16     | DATA<31:16>         |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | DATA<15:0>          |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7E00            | ADCSYSCFG0               | 31:16     | —                   | —                   | —                   | —     | AN27  | AN26  | AN25                | AN24                | AN23 <sup>(1)</sup> | AN22 <sup>(1)</sup> | AN21 <sup>(1)</sup> | AN20 <sup>(1)</sup> | AN19                | AN18                | AN17                | AN16       | 0FxF |
|                 |                          | 15:0      | AN15                | AN14                | AN13                | AN12  | AN11  | AN10  | AN9                 | AN8                 | AN7                 | AN6                 | AN5                 | AN4                 | AN3                 | AN2                 | AN1                 | AN0        | FFFF |
| 7E10            | ADCSYSCFG1               | 31:16     | —                   | —                   | —                   | —     | —     | —     | —                   | —                   | —                   | AN53 <sup>(1)</sup> | AN52 <sup>(1)</sup> | —                   | AN50 <sup>(1)</sup> | AN49                | AN48                | 00xx       |      |
|                 |                          | 15:0      | AN47 <sup>(1)</sup> | AN46 <sup>(1)</sup> | AN45 <sup>(1)</sup> | —     | —     | —     | AN41 <sup>(1)</sup> | AN40 <sup>(1)</sup> | AN39 <sup>(1)</sup> | AN38 <sup>(1)</sup> | AN37 <sup>(1)</sup> | AN36 <sup>(1)</sup> | AN35 <sup>(1)</sup> | AN34 <sup>(1)</sup> | AN33 <sup>(1)</sup> | —          | xxxx |
| 7D00            | ADC0CFG <sup>(3)</sup>   | 31:16     | ADCCFG<31:16>       |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | ADCCFG<15:0>        |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
| 7D10            | ADC1CFG <sup>(3)</sup>   | 31:16     | ADCCFG<31:16>       |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |
|                 |                          | 15:0      | ADCCFG<15:0>        |                     |                     |       |       |       |                     |                     |                     |                     |                     |                     |                     |                     |                     | 0000       |      |

**Note**

- 1: This bit or register is not available on 64-pin devices.
- 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
- 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.



**TABLE 23-2: ADC REGISTER MAP (CONTINUED)**

| Virtual Address | Register Name          | Bit Range | Bits          |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets |
|-----------------|------------------------|-----------|---------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------------|
|                 |                        |           | 31/15         | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |            |
| 7D20            | ADC2CFG <sup>(3)</sup> | 31:16     | ADCCFG<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                        | 15:0      | ADCCFG<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7D30            | ADC3CFG <sup>(3)</sup> | 31:16     | ADCCFG<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                        | 15:0      | ADCCFG<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7D40            | ADC4CFG <sup>(3)</sup> | 31:16     | ADCCFG<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                        | 15:0      | ADCCFG<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7D50            | ADC5CFG <sup>(3)</sup> | 31:16     | ADCCFG<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                        | 15:0      | ADCCFG<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
| 7D70            | ADC7CFG <sup>(3)</sup> | 31:16     | ADCCFG<31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |
|                 |                        | 15:0      | ADCCFG<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | 0000       |

- Note**
- 1: This bit or register is not available on 64-pin devices.
  - 2: These register bits are for internal ADC input sources (i.e., IVREF, and CTMU Temperature Sensor).
  - 3: Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-1: ADCCON1: ADC CONTROL REGISTER 1

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6       | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2    | Bit 25/17/9/1      | Bit 24/16/8/0 |
|-----------|----------------|----------------------|----------------|----------------|----------------|-------------------|--------------------|---------------|
| 31:24     | U-0<br>—       | U-0<br>—             | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—          | U-0<br>—           | U-0<br>—      |
| 23:16     | R/W-0<br>FRACT | R/W-1<br>SELRES<1:0> | R/W-1          | R/W-0          | R/W-0          | STRGSRC<4:0>      |                    |               |
| 15:8      | R/W-0<br>ON    | U-0                  | R/W-0<br>SIDL  | U-0<br>—       | R/W-0<br>CVDEN | R/W-0<br>FSSCLKEN | R/W-0<br>FSPBCLKEN | U-0<br>—      |
| 7:0       | U-0<br>—       | R/W-0                | R/W-0          | R/W-0          | R/W-0          | R/W-0             | R/W-0              | R/W-0         |
|           |                | IRQVS<2:0>           |                |                | STRGLVL        | DMABL<2:0>        |                    |               |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HC = Hardware Set | HS = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-24 **Unimplemented:** Read/Write as '0'

bit 23 **FRACT:** Fractional Data Output Format bit

1 = Fractional

0 = Integer

bit 22-21 **SELRES<1:0>:** Shared ADC7 (i.e., AN6-AN53) Resolution bits

11 = 12 bits (default)

10 = 10 bits

01 = 8 bits

00 = 6 bits

**Note 1:** The PWM bit definitions are only applicable to the PIC32MKXXXMCJXX Motor Control devices otherwise these bits are reserved.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-1: ADCCON1: ADC CONTROL REGISTER 1 (CONTINUED)

bit 20-16 **STRGSRC<4:0>**: Scan Trigger Source Select bits

11111 = Reserved  
11110 = PWM Generator 12 trigger<sup>(1)</sup>  
11101 = PWM Generator 11 trigger<sup>(1)</sup>  
11100 = PWM Generator 10 trigger<sup>(1)</sup>  
11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
11001 = PWM Generator 2 Current-Limit  
11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
10111 = PWM Generator 9 trigger<sup>(1)</sup>  
10110 = PWM Generator 8 trigger<sup>(1)</sup>  
10101 = PWM Generator 7 trigger<sup>(1)</sup>  
10100 = CTMU trip  
10011 = Output Compare 4 period end  
10010 = Output Compare 3 period end  
10001 = Output Compare 2 period end  
10000 = Output Compare 1 period end  
01111 = PWM Generator 6 trigger<sup>(1)</sup>  
01110 = PWM Generator 5 trigger<sup>(1)</sup>  
01101 = PWM Generator 4 trigger<sup>(1)</sup>  
01100 = PWM Generator 3 trigger<sup>(1)</sup>  
01011 = PWM Generator 2 trigger<sup>(1)</sup>  
01010 = PWM Generator 1 trigger<sup>(1)</sup>  
01001 = Secondary PWM time base<sup>(1)</sup>  
01000 = Primary PWM time base<sup>(1)</sup>  
00111 = General Purpose Timer5  
00110 = General Purpose Timer3  
00101 = General Purpose Timer1  
00100 = INT0  
00011 = Scan trigger  
00010 = Software level trigger  
00001 = Software edge trigger  
00000 = No Trigger

**Note:** These triggers only apply to implemented analog inputs AN32-AN53. For AN0-AN27 refer to ADCTRG1-ADCTRG7.

bit 15 **ON**: ADC Module Enable bit

1 = ADC module is enabled  
0 = ADC module is disabled

**Note:** The ON bit should be set only after the ADC module has been configured.

bit 14 **Unimplemented**: Read as '0'

bit 13 **SIDL**: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode  
0 = Continue module operation in Idle mode

bit 12 **Reserved**: (Read/Write as 0)

bit 11 **CVDEN**: Capacitive Voltage Division Enable bit

1 = CVD operation is enabled  
0 = CVD operation is disabled

bit 10 **FSSCLKEN**: Bypass Fast Synchronous DMA System Clock to ADC Control Clock

1 = Bypass synchronizer logic for DMA system clock to ADC control clocks  
0 = Enable clock synchronizers for non-synchronized DMA to ADC clock sources

**NOTE:** Synchronizers required if ADCCON3<ADCSEL> = REFCLK3, or ADCCON3<ADCSEL> = FRC and FRC is not SYSCLK source otherwise this bit is n/a.

**Note 1:** The PWM bit definitions are only applicable to the PIC32MKXXXMCJXX Motor Control devices otherwise these bits are reserved.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-1: ADCCON1: ADC CONTROL REGISTER 1 (CONTINUED)

- bit 9 **FSPBCLKEN:** Bypass Fast Synchronous Peripheral Bus Clock to ADC Control Clock  
1 = Bypass synchronizer logic for peripheral clock to ADC control clocks  
0 = Enable clock synchronizers for non-synchronized peripheral clock to ADC control clocks  
**NOTE:** Synchronizers required if ADCCON3<ADCSEL> = REFCLK3, or ADCCON3<ADCSEL> = FRC and FRC is not SYSCLK source otherwise this bit is n/a.
- bit 8-7 **Unimplemented:** Read as '0'
- bit 6-4 **IRQVS<2:0>:** Interrupt Vector Shift bits  
To determine interrupt vector address, this bit specifies the amount of left shift done to the AIRDYx status bits in the ADCDSTAT1 and ADCDSTAT2 registers, prior to adding with the ADCBASE register.  
Interrupt Vector Address = Read Value of ADCBASE and Read Value of ADCBASE = Value written to ADCBASE + x << IRQVS<2:0>, where 'x' is the smallest active input ID from the ADCDSTAT1 or ADCDSTAT2 registers (which has highest priority).  
111 = Shift x left 7 bit position  
110 = Shift x left 6 bit position  
101 = Shift x left 5 bit position  
100 = Shift x left 4 bit position  
011 = Shift x left 3 bit position  
010 = Shift x left 2 bit position  
001 = Shift x left 1 bit position  
000 = Shift x left 0 bit position
- bit 3 **STRGLVL:** Scan Trigger High Level/Positive Edge Sensitivity bit  
1 = Scan trigger is high level sensitive. Once STRIG mode is selected (TRGSRCx<4:0> in the ADCTRGx register), the scan trigger will continue for all selected analog inputs, until the STRIG option is removed.  
0 = Scan trigger is positive edge sensitive. Once STRIG mode is selected (TRGSRCx<4:0> in the ADCTRGx register), only a single scan trigger will be generated, which will complete the scan of all selected analog inputs.
- bit 2-0 **DMABL<2:0>:** DMA to System RAM Buffer Length Size bits  
These bits define the number of locations in system memory allocated per analog input for DMA interface use.  
Because each output data is 16-bit wide, one location consists of 2 bytes. Therefore the actual size reserved in the System RAM follows the formula: RAM Buffer Length in bytes = 2(DMABL+1).  
The DMABL field can also be thought of as a "Left Shift Amount +1" needed for the channel ID to create the DMA byte address offset to be added to the contents of ADDMAB in order to obtain the byte address of the beginning of the System RAM buffer area allocated for the given channel.  
111 = Allocates 128 locations in system memory to each analog input, actually 256 bytes  
110 = Allocates 64 locations in system memory to each analog input, actually 128 bytes  
101 = Allocates 32 locations in system memory to each analog input, actually 64 bytes  
100 = Allocates 16 locations in system memory to each analog input, actually 32 bytes  
011 = Allocates 8 locations in system memory to each analog input, actually 16 bytes  
010 = Allocates 4 locations in system memory to each analog input, actually 8 bytes  
001 = Allocates 2 locations in system memory to each analog input, actually 4 bytes  
000 = Allocates 1 location in system memory to each analog input, actually 2 bytes
- Note 1:** The PWM bit definitions are only applicable to the PIC32MKXXXMCJXX Motor Control devices otherwise these bits are reserved.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-2: ADCCON2: ADC CONTROL REGISTER 2**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | BGVERRDY          | REFFLT            | EOSRDY            | CVDCPL<2:0>       |                   |                   | SAMC<9:8>        |                  |
| 23:16     | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | SAMC<7:0>         |                   |                   |                   |                   |                   |                  |                  |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | U-0               | R/W-0             | R/W-0            | R/W-0            |
|           | BGVRIEN           | REFFLTIEN         | EOSIEN            | ADCEIOVR          | —                 | ADCEIS<2:0>       |                  |                  |
| 7:0       | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | ADCDIV<6:0>       |                   |                   |                   |                   |                  |                  |

|                   |                   |                                    |                    |
|-------------------|-------------------|------------------------------------|--------------------|
| <b>Legend:</b>    | HC = Hardware Set | HS = Hardware Cleared              | r = Reserved       |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0' |                    |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared               | x = Bit is unknown |

- bit 31 **BGVERRDY:** Band Gap Voltage/ADC Reference Voltage Status bit  
 1 = Both band gap voltage and ADC reference voltages (VREF) are ready  
 0 = Either or both band gap voltage and ADC reference voltages (VREF) are not ready  
 Data processing is valid only after BGVERRDY is set by hardware, so the application code must check that the BGVERRDY bit is set to ensure data validity. This bit set to '0' when ON (ADCCON1<15>) = 0.
- bit 30 **REFFLT:** Band Gap/VREF/AVDD BOR Fault Status bit  
 1 = Fault in band gap or the VREF voltage while the ON bit (ADCCON1<15>) was set. Most likely, a band gap or VREF fault will be caused by a BOR of the analog VDD supply.  
 0 = Band gap and VREF voltage are working properly  
 This bit is cleared when the ON bit (ADCCON1<15>) = 0 and the BGVERRDY bit = 1.
- bit 29 **EOSRDY:** End of Scan Interrupt Status bit  
 1 = All analog inputs are considered for scanning through the scan trigger (all analog inputs specified in the ADCCSS1 and ADCCSS2 registers) have completed scanning  
 0 = Scanning has not completed  
 This bit is cleared when ADCCON2<31:24> are read in software.
- bit 28-26 **CVDCPL<2:0>:** Capacitor Voltage Divider (CVD) Setting bits  
 111 = 7 \* 2.5 pF = 17.5 pF  
 110 = 6 \* 2.5 pF = 15 pF  
 101 = 5 \* 2.5 pF = 12.5 pF  
 100 = 4 \* 2.5 pF = 10 pF  
 011 = 3 \* 2.5 pF = 7.5 pF  
 010 = 2 \* 2.5 pF = 5 pF  
 001 = 1 \* 2.5 pF = 2.5 pF  
 000 = 0 \* 2.5 pF = 0 pF  
**Note:** These bits are available only on shared ADC7 inputs AN6-AN49. Once enabled (CVD-CPL<2:0> > 000), the internal capacitors are internally connected to all ADC7 inputs. To determine user ADC sampling time requirements (SAMC<9:0> bits (ADCCON2<25:16>)) with CVD-CPL selection, refer to [TABLE 36-41: "ADC Sample Times with CVD Enabled"](#).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-2: ADCCON2: ADC CONTROL REGISTER 2 (CONTINUED)

bit 25-16 **SAMC<9:0>**: Sample Time for the Shared ADC (ADC7) bits

1111111111 = 1025 TAD

.

.

.

0000000001 = 3 TAD

0000000000 = 2 TAD

Where TAD = period of the ADC conversion clock for the Shared ADC (ADC7) controlled by the ADCDIV<6:0> bits.

Shared ADC7 Throughput rate:

$= ((1/((\text{Sample time} + \text{Conversion time})(\text{TAD}))) / \text{\#of ADC inputs used in scan list})$

$= ((1 / ((\text{SAMC} + \text{\# bit resolution} + 1)(\text{TAD}))) / \text{\#of ADC inputs used in scan list})$

*Example:*

SCAN mode enabled with (2) ANx inputs in scan list, (i.e. ADCCSSx<CSSy>), SAMC = 4 TAD, 12-bit mode, TAD = 16.667ns = 60 MHz:

Throughput rate =  $((1/((4+12+1)(16.667 \text{ ns}))) / 2)$

$= ((1/(17 * 16.667 \text{ ns})) / 2)$

$= 1.764706 \text{ msp}$

**Note:** Unlike the High-Speed Class 1 ADC modules, the trigger event for the shared Class 3 ADC7 module initiates the SAMC *sampling* sequence, rather than the *convert* sequence.

bit 15 **BGVRIEN**: Band Gap/VREF Voltage Ready Interrupt Enable bit

1 = Interrupt will be generated when the BGVRRDY bit is set

0 = No interrupt is generated when the BGVRRDY bit is set

bit 14 **REFFLTIEN**: Band Gap/VREF Voltage Fault Interrupt Enable bit

1 = Interrupt will be generated when the REFFLT bit is set

0 = No interrupt is generated when the REFFLT bit is set

bit 13 **EOSIEN**: End of Scan Interrupt Enable bit

1 = Interrupt will be generated when EOSRDY bit is set

0 = No interrupt is generated when the EOSRDY bit is set

bit 12 **ADCEIOVR**: Early Interrupt Request Override bit

1 = Early interrupt generation is overridden and interrupt generation is controlled by the ADCGIRQEN1 and ADCGIRQEN2 registers

0 = Early interrupt generation is not overridden and interrupt generation is controlled by the ADCEIEN1 and ADCEIEN2 registers

bit 11 **Unimplemented**: Read as '0'

bit 10-8 **ADCEIS<2:0>**: Shared ADC (ADC7) Early Interrupt Select bits

These bits select the number of clocks (TAD7) prior to the arrival of valid data that the associated interrupt is generated.

111 = The data ready interrupt is generated 8 ADC clocks prior to end of conversion

110 = The data ready interrupt is generated 7 ADC clocks prior to end of conversion

.

.

.

001 = The data ready interrupt is generated 2 ADC module clocks prior to end of conversion

000 = The data ready interrupt is generated 1 ADC module clock prior to end of conversion

**Note:** All options are available when the selected resolution, set by the SELRES<1:0> bits (ADCCON1<22:21>), is 12-bit or 10-bit. For a selected resolution of 8-bit, options from '000' to '101' are valid. For a selected resolution of 6-bit, options from '000' to '011' are valid.

bit 7 **Unimplemented**: Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-2: ADCCON2: ADC CONTROL REGISTER 2 (CONTINUED)

bit 6-0      **ADCDIV<6:0>**: Shared ADC (ADC7) Clock Divider bits

11111111 = 254 \* TQ = TAD

.

.

00000111 = 6 \* TQ = TAD

00000101 = 4 \* TQ = TAD

00000011 = 2 \* TQ = TAD

00000000 = Reserved

The ADCDIV<6:0> bits divide the ADC control clock (TQ) to generate the clock for the Shared ADC, ADC7 (TAD7).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-3: ADCCON3: ADC CONTROL REGISTER 3

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1             | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0                     | R/W-0         |
|           | ADCSEL<1:0>    |                |                | CONCLKDIV<5:0> |                |                |                           |               |
| 23:16     | R/W-0          | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0                     | R/W-0         |
|           | DIGEN7         | —              | DIGEN5         | DIGEN4         | DIGEN3         | DIGEN2         | DIGEN1                    | DIGEN0        |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R-0, HS, HC    | R/W-0                     | R-0, HS, HC   |
|           | VREFSEL<2:0>   |                |                | TRGSUSP        | UPDIEN         | UPDRDY         | SAMP <sup>(1,2,3,4)</sup> | RQCNVRT       |
| 7:0       | R/W-0          | R-0, HS, HC    | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0                     | R/W-0         |
|           | GLSWTRG        | GSWTRG         | ADINSEL<5:0>   |                |                |                |                           |               |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HC = Hardware Set | HS = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-30 **ADCSEL<1:0>**: Analog-to-Digital Clock Source (TCLK) bits

- 11 = SYSCLK (Required if using DMA for ADC)
- 10 = REFCLK3
- 01 = FRC
- 00 = SYSCLK

bit 29-24 **CONCLKDIV<5:0>**: Analog-to-Digital Control Clock (Tq) Divider bits

- 111111 = 126 \* TCLK = Tq
- .
- .
- .
- 000011 = 6 \* TCLK = Tq
- 000010 = 4 \* TCLK = Tq
- 000001 = 2 \* TCLK = Tq
- 000000 = TCLK = Tq

bit 23 **DIGEN7**: Shared ADC (ADC7) Digital Enable bit

- 1 = ADC7 is digital enabled
- 0 = ADC7 is digital disabled

bit 22 **Unimplemented**: Read as '0'

bit 21 **DIGEN5**: ADC5 Digital Enable bit

- 1 = ADC5 is digital enabled (required for active operation)
- 0 = ADC5 is digital disabled (power-saving mode)

bit 20 **DIGEN4**: ADC4 Digital Enable bit

- 1 = ADC4 is digital enabled (required for active operation)
- 0 = ADC4 is digital disabled (power-saving mode)

**Note 1:** The SAMP bit has the highest priority and setting this bit will keep the S&H circuit in Sample mode until the bit is cleared. Also, usage of the SAMP bit will cause settings of SAMC<9:0> bits (ADCCON2<25:16>) to be ignored.

- 2: The SAMP bit only connects analog inputs to the shared ADC, ADC7. All Class 1 analog inputs are not affected by the SAMP bit.
- 3: The SAMP bit is not a self-clearing bit and it is the responsibility of application software to first clear this bit and only after setting the RQCNVRT bit to start the analog-to-digital conversion.
- 4: Normally, when the SAMP and RQCNVRT bits are used by software routines, all TRGSRCx<4:0> bits and STRGSRC<4:0> bits should be set to '00000' to disable all external hardware triggers and prevent them from interfering with the software-controlled sampling command signal SAMP and with the software-controlled trigger RQCNVRT.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-3: ADCCON3: ADC CONTROL REGISTER 3 (CONTINUED)

bit 19 **DIGEN3:** ADC3 Digital Enable bit  
 1 = ADC3 is digital enabled (required for active operation)  
 0 = ADC3 is digital disabled (power-saving mode)

bit 18 **DIGEN2:** ADC2 Digital Enable bit  
 1 = ADC2 is digital enabled (required for active operation)  
 0 = ADC2 is digital disabled (power-saving mode)

bit 17 **DIGEN1:** ADC1 Digital Enable bit  
 1 = ADC1 is digital enabled (required for active operation)  
 0 = ADC1 is digital disabled (power-saving mode)

bit 16 **DIGEN0:** ADC0 Digital Enable bit  
 1 = ADC0 is digital enabled (required for active operation)  
 0 = ADC0 is digital disabled (power-saving mode)

bit 15-13 **VREFSEL<2:0>:** Voltage Reference (VREF) Input Selection bits

| VREFSEL<2:0> | ADC VREFH | ADC VREFL |
|--------------|-----------|-----------|
| 1xx          | Reserved  | Reserved  |
| 011          | VREF+     | VREF-     |
| 010          | AVDD      | VREF-     |
| 001          | VREF+     | AVSS      |
| 000          | AVDD      | AVSS      |

bit 12 **TRGSUSP:** Trigger Suspend bit  
 1 = Triggers are blocked from starting a new analog-to-digital conversion, but the ADC module is not disabled  
 0 = Triggers are not blocked

bit 11 **UPDIEN:** Update Ready Interrupt Enable bit  
 1 = Interrupt will be generated when the UPDRDY bit is set by hardware  
 0 = No interrupt is generated

bit 10 **UPDRDY:** ADC Update Ready Status bit  
 1 = ADC SFRs can be updated  
 0 = ADC SFRs cannot be updated

**Note:** This bit is only active while the TRGSUSP bit is set and there are no more running conversions of any ADC modules.

bit 9 **SAMP:** Shared ADC7 Analog Input Sampling Enable bit<sup>(1,2,3,4)</sup>  
 1 = The ADC S&H amplifier is sampling  
 0 = The ADC S&H amplifier is holding

bit 8 **RQCNVRT:** Individual ADC Input Conversion Request bit  
 This bit and its associated ADINSEL<5:0> bits enable the user to individually request an analog-to-digital conversion of an analog input through software.  
 1 = Trigger the conversion of the selected ADC input as specified by the ADINSEL<5:0> bits  
 0 = Do not trigger the conversion  
**Note:** This bit is automatically cleared in the next ADC clock cycle.

**Note 1:** The SAMP bit has the highest priority and setting this bit will keep the S&H circuit in Sample mode until the bit is cleared. Also, usage of the SAMP bit will cause settings of SAMC<9:0> bits (ADCCON2<25:16>) to be ignored.

**2:** The SAMP bit only connects analog inputs to the shared ADC, ADC7. All Class 1 analog inputs are not affected by the SAMP bit.

**3:** The SAMP bit is not a self-clearing bit and it is the responsibility of application software to first clear this bit and only after setting the RQCNVRT bit to start the analog-to-digital conversion.

**4:** Normally, when the SAMP and RQCNVRT bits are used by software routines, all TRGSRCx<4:0> bits and STRGSRC<4:0> bits should be set to '00000' to disable all external hardware triggers and prevent them from interfering with the software-controlled sampling command signal SAMP and with the software-controlled trigger RQCNVRT.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-3: ADCCON3: ADC CONTROL REGISTER 3 (CONTINUED)

- bit 7     **GLSWTRG:** Global Level Software Trigger bit  
1 = Trigger conversion for ADC inputs that have selected the GLSWTRG bit as the trigger signal, either through the associated TRGSRC<4:0> bits in the ADCTRGx registers or through the STRGSRC<4:0> bits in the ADCCON1 register  
0 = Do not trigger an analog-to-digital conversion
- bit 6     **GSWTRG:** Global Software Trigger bit  
1 = Trigger conversion for ADC inputs that have selected the GSWTRG bit as the trigger signal, either through the associated TRGSRC<4:0> bits in the ADCTRGx registers or through the STRGSRC<4:0> bits in the ADCCON1 register  
0 = Do not trigger an analog-to-digital conversion  
**Note:** This bit is automatically cleared in the next ADC clock cycle.

- Note 1:** The SAMP bit has the highest priority and setting this bit will keep the S&H circuit in Sample mode until the bit is cleared. Also, usage of the SAMP bit will cause settings of SAMC<9:0> bits (ADCCON2<25:16>) to be ignored.
- 2:** The SAMP bit only connects analog inputs to the shared ADC, ADC7. All Class 1 analog inputs are not affected by the SAMP bit.
- 3:** The SAMP bit is not a self-clearing bit and it is the responsibility of application software to first clear this bit and only after setting the RQCNVRT bit to start the analog-to-digital conversion.
- 4:** Normally, when the SAMP and RQCNVRT bits are used by software routines, all TRGSRCx<4:0> bits and STRGSRC<4:0> bits should be set to '00000' to disable all external hardware triggers and prevent them from interfering with the software-controlled sampling command signal SAMP and with the software-controlled trigger RQCNVRT.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-3: ADCCON3: ADC CONTROL REGISTER 3 (CONTINUED)

bit 5-0 **ADINSEL<5:0>**: Analog Input Select bits

These bits select the analog input to be converted when the RQCNVRT bit is set.

111111 = Reserved  
.  
.  
.  
110110 = Reserved  
110101 = CTMU Temperature Sensor (internal AN53)  
110100 = Reserved  
110011 = Reserved  
110010 = IVREF 1.2V (internal AN50)  
110001 = AN49  
.  
.  
.  
101101 = AN45  
101100 = Reserved  
.  
.  
.  
101010 = Reserved  
101001 = AN41  
.  
.  
.  
100001 = AN33  
100000 = Reserved  
.  
.  
.  
011100 = Reserved  
011011 = AN27  
.  
.  
.  
000000 = AN0

**Note:** AN20-AN23, AN33-AN41, and AN45-AN47 are not available on 64-pin devices. Refer to [TABLE 1-1: "PORTA THrough PORTG REMAPPABLE PERIPHERAL DESCRIPTIONS"](#) for details.

- Note 1:** The SAMP bit has the highest priority and setting this bit will keep the S&H circuit in Sample mode until the bit is cleared. Also, usage of the SAMP bit will cause settings of SAMC<9:0> bits (ADCCON2<25:16>) to be ignored.
- 2:** The SAMP bit only connects analog inputs to the shared ADC, ADC7. All Class 1 analog inputs are not affected by the SAMP bit.
  - 3:** The SAMP bit is not a self-clearing bit and it is the responsibility of application software to first clear this bit and only after setting the RQCNVRT bit to start the analog-to-digital conversion.
  - 4:** Normally, when the SAMP and RQCNVRT bits are used by software routines, all TRGSRCx<4:0> bits and STRGSRC<4:0> bits should be set to '00000' to disable all external hardware triggers and prevent them from interfering with the software-controlled sampling command signal SAMP and with the software-controlled trigger RQCNVRT.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-4: ADCTRGMODE: ADC TRIGGERING MODE FOR DEDICATED ADC REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | SH5ALT<1:0>    |                | SH4ALT<1:0>   |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | SH3ALT<1:0>    |                | SH2ALT<1:0>    |                | SH1ALT<1:0>    |                | SH0ALT<1:0>   |               |
| 15:8      | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | STRGEN5        | STRGEN4        | STRGEN3        | STRGEN2        | STRGEN1       | STRGEN0       |
| 7:0       | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | SSAMPEN5       | SSAMPEN4       | SSAMPEN3       | SSAMPEN2       | SSAMPEN1      | SSAMPEN0      |

### Legend:

R = Readable bit    W = Writable bit    U = Unimplemented bit, read as '0'  
 -n = Value at POR    '1' = Bit is set    '0' = Bit is cleared    x = Bit is unknown

bit 31-28 **Unimplemented:** Read as '0'

bit 27-26 **SH5ALT<1:0>**: ADC5 Analog Input Select bit

11 = AN25<sup>(1)</sup>  
 10 = AN6<sup>(1)</sup>  
 01 = AN2<sup>(1)</sup>  
 00 = AN5

bit 25-24 **SH4ALT<1:0>**: ADC4 Analog Input Select bit

11 = AN0<sup>(1)</sup>  
 10 = AN9<sup>(1)</sup>  
 01 = AN1<sup>(1)</sup>  
 00 = AN4

bit 23-22 **SH3ALT<1:0>**: ADC3 Analog Input Select bit

11 = AN26<sup>(1)</sup>  
 10 = AN8<sup>(1)</sup>  
 01 = AN0<sup>(1)</sup>  
 00 = AN3

bit 21-20 **SH2ALT<1:0>**: ADC2 Analog Input Select bit

11 = AN25<sup>(1)</sup>  
 10 = AN6<sup>(1)</sup>  
 01 = AN5<sup>(1)</sup>  
 00 = AN2

bit 19-18 **SH1ALT<1:0>**: ADC1 Analog Input Select bit

11 = AN0<sup>(1)</sup>  
 10 = AN7<sup>(1)</sup>  
 01 = AN4<sup>(1)</sup>  
 00 = AN1

bit 17-16 **SH0ALT<1:0>**: ADC0 Analog Input Select bit

11 = AN24<sup>(1)</sup>  
 10 = AN5<sup>(1)</sup>  
 01 = AN3<sup>(1)</sup>  
 00 = AN0

**Note 1:** Regardless of what alternate input is selected by SHxALT, only for ADC0-ADC5, all control and results are handled by the native SHxALT = '0b00' input. For example, SH0ALT = '0b11' = AN24. However, from a software and silicon hardware control and results register perspective, the user application must initialize the ADC0 module as if AN24 were actually AN0.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-4: ADCTRGMODE: ADC TRIGGERING MODE FOR DEDICATED ADC REGISTER (CONTINUED)

bit 15-14 **Unimplemented:** Read as '0'

- bit 13 **STRGEN5:** ADC5 Presynchronized Triggers bit  
1 = ADC5 uses presynchronized triggers  
0 = ADC5 does not use presynchronized triggers
- bit 12 **STRGEN4:** ADC4 Presynchronized Triggers bit  
1 = ADC4 uses presynchronized triggers  
0 = ADC4 does not use presynchronized triggers
- bit 11 **STRGEN3:** ADC3 Presynchronized Triggers bit  
1 = ADC3 uses presynchronized triggers  
0 = ADC3 does not use presynchronized triggers
- bit 10 **STRGEN2:** ADC2 Presynchronized Triggers bit  
1 = ADC2 uses presynchronized triggers  
0 = ADC2 does not use presynchronized triggers
- bit 9 **STRGEN1:** ADC1 Presynchronized Triggers bit  
1 = ADC1 uses presynchronized triggers  
0 = ADC1 does not use presynchronized triggers
- bit 8 **STRGEN0:** ADC0 Presynchronized Triggers bit  
1 = ADC0 uses presynchronized triggers  
0 = ADC0 does not use presynchronized triggers
- bit 7-6 **Unimplemented:** Read as '0'
- bit 5 **SSAMPEN5:** ADC5 Synchronous Sampling bit  
1 = ADC5 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC5 does not use synchronous sampling
- bit 4 **SSAMPEN4:** ADC4 Synchronous Sampling bit  
1 = ADC4 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC4 does not use synchronous sampling
- bit 3 **SSAMPEN3:** ADC3 Synchronous Sampling bit  
1 = ADC3 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC3 does not use synchronous sampling
- bit 2 **SSAMPEN2:** ADC2 Synchronous Sampling bit  
1 = ADC2 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC2 does not use synchronous sampling
- bit 1 **SSAMPEN1:** ADC1 Synchronous Sampling bit  
1 = ADC1 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC1 does not use synchronous sampling
- bit 0 **SSAMPEN0:** ADC0 Synchronous Sampling bit  
1 = ADC0 uses synchronous sampling for the first sample after being idle or disabled  
0 = ADC0 does not use synchronous sampling

**Note 1:** Regardless of what alternate input is selected by SHxALT, only for ADC0-ADC5, all control and results are handled by the native SHxALT = '0b00' input. For example, SH0ALT = '0b11' = AN24. However, from a software and silicon hardware control and results register perspective, the user application must initialize the ADC0 module as if AN24 were actually AN0.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-5: ADCIMCON1: ADC INPUT MODE CONTROL REGISTER 1**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DIFF15            | SIGN15            | DIFF14            | SIGN14            | DIFF13            | SIGN13            | DIFF12           | SIGN12           |
| 23:16     | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DIFF11            | SIGN11            | DIFF10            | SIGN10            | DIFF9             | SIGN9             | DIFF8            | SIGN8            |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DIFF7             | SIGN7             | DIFF6             | SIGN6             | DIFF5             | SIGN5             | DIFF4            | SIGN4            |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DIFF3             | SIGN3             | DIFF2             | SIGN2             | DIFF1             | SIGN1             | DIFF0            | SIGN0            |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

- bit 31      **DIFF15:** AN15 Mode bit  
             1 = Selects AN15 differential input pair as AN15+ and AN1-  
             0 = AN15 is using Single-ended mode
- bit 30      **SIGN:15** AN15 Signed Data Mode bit  
             1 = AN15 is using Signed Data mode  
             0 = AN15 is using Unsigned Data mode
- bit 29      **DIFF14:** AN14 Mode bit  
             1 = Selects AN14 differential input pair as AN14+ and AN1-  
             0 = AN14 is using Single-ended mode
- bit 28      **SIGN14:** AN14 Signed Data Mode bit  
             1 = AN14 is using Signed Data mode  
             0 = AN14 is using Unsigned Data mode
- bit 27      **DIFF13:** AN13 Mode bit  
             1 = Selects AN13 differential input pair as AN13+ and AN1-  
             0 = AN13 is using Single-ended mode
- bit 26      **SIGN13:** AN13 Signed Data Mode bit  
             1 = AN13 is using Signed Data mode  
             0 = AN13 is using Unsigned Data mode
- bit 25      **DIFF12:** AN12 Mode bit  
             1 = Selects AN12 differential input pair as AN12+ and AN1-  
             0 = AN12 is using Single-ended mode
- bit 24      **SIGN12:** AN12 Signed Data Mode bit  
             1 = AN12 is using Signed Data mode  
             0 = AN12 is using Unsigned Data mode
- bit 23      **DIFF11:** AN11 Mode bit  
             1 = Selects AN11 differential input pair as AN11+ and AN1-  
             0 = AN11 is using Single-ended mode
- bit 22      **SIGN11:** AN11 Signed Data Mode bit  
             1 = AN11 is using Signed Data mode  
             0 = AN11 is using Unsigned Data mode
- bit 21      **DIFF10:** AN10 Mode bit  
             1 = Selects AN10 differential input pair as AN10+ and AN1-  
             0 = AN10 is using Single-ended mode

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-5: ADCIMCON1: ADC INPUT MODE CONTROL REGISTER 1 (CONTINUED)

|        |   |
|--------|---|
| bit 20 | <b>SIGN10:</b> AN10 Signed Data Mode bit<br>1 = AN10 is using Signed Data mode<br>0 = AN10 is using Unsigned Data mode        |
| bit 19 | <b>DIFF9:</b> AN9 Mode bit<br>1 = Selects AN9 differential input pair as AN9+ and AN1-<br>0 = AN9 is using Single-ended mode  |
| bit 18 | <b>SIGN9:</b> AN9 Signed Data Mode bit<br>1 = AN9 is using Signed Data mode<br>0 = AN9 is using Unsigned Data mode            |
| bit 17 | <b>DIFF8:</b> AN 8 Mode bit<br>1 = Selects AN8 differential input pair as AN8+ and AN1-<br>0 = AN8 is using Single-ended mode |
| bit 16 | <b>SIGN8:</b> AN8 Signed Data Mode bit<br>1 = AN8 is using Signed Data mode<br>0 = AN8 is using Unsigned Data mode            |
| bit 15 | <b>DIFF7:</b> AN7 Mode bit<br>1 = Selects AN7 differential input pair as AN7+ and AN1-<br>0 = AN7 is using Single-ended mode  |
| bit 14 | <b>SIGN7:</b> AN7 Signed Data Mode bit<br>1 = AN7 is using Signed Data mode<br>0 = AN7 is using Unsigned Data mode            |
| bit 13 | <b>DIFF6:</b> AN6 Mode bit<br>1 = Selects AN6 differential input pair as AN6+ and AN1-<br>0 = AN6 is using Single-ended mode  |
| bit 12 | <b>SIGN6:</b> AN6 Signed Data Mode bit<br>1 = AN6 is using Signed Data mode<br>0 = AN6 is using Unsigned Data mode            |
| bit 11 | <b>DIFF5:</b> AN5 Mode bit<br>1 = Selects AN5 differential input pair as AN5+ and AN11-<br>0 = AN5 is using Single-ended mode |
| bit 10 | <b>SIGN5:</b> AN5 Signed Data Mode bit<br>1 = AN5 is using Signed Data mode<br>0 = AN5 is using Unsigned Data mode            |
| bit 9  | <b>DIFF4:</b> AN4 Mode bit<br>1 = Selects AN4 differential input pair as AN4+ and AN10-<br>0 = AN4 is using Single-ended mode |
| bit 8  | <b>SIGN4:</b> AN4 Signed Data Mode bit<br>1 = AN4 is using Signed Data mode<br>0 = AN4 is using Unsigned Data mode            |
| bit 7  | <b>DIFF3:</b> AN3 Mode bit<br>1 = Selects AN3 differential input pair as AN3+ and AN27-<br>0 = AN3 is using Single-ended mode |
| bit 6  | <b>SIGN3:</b> AN3 Signed Data Mode bit<br>1 = AN3 is using Signed Data mode<br>0 = AN3 is using Unsigned Data mode            |
| bit 5  | <b>DIFF2:</b> AN2 Mode bit<br>1 = Selects AN2 differential input pair as AN2+ and AN8-<br>0 = AN2 is using Single-ended mode  |
| bit 4  | <b>SIGN2:</b> AN2 Signed Data Mode bit<br>1 = AN2 is using Signed Data mode<br>0 = AN2 is using Unsigned Data mode            |

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-5: ADCIMCON1: ADC INPUT MODE CONTROL REGISTER 1 (CONTINUED)

- bit 3      **DIFF1:** AN1 Mode bit  
1 = Selects AN1 differential input pair as AN1+ and AN7-  
0 = AN1 is using Single-ended mode
- bit 2      **SIGN1:** AN1 Signed Data Mode bit  
1 = AN1 is using Signed Data mode  
0 = AN1 is using Unsigned Data mode
- bit 1      **DIFF0:** AN0 Mode bit  
1 = Selects AN0 differential input pair as AN0+ and AN6-  
0 = AN0 is using Single-ended mode
- bit 0      **SIGN0:** AN0 Signed Data Mode bit  
1 = AN0 is using Signed Data mode  
0 = AN0 is using Unsigned Data mode



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-6: ADCIMCON2: ADC INPUT MODE CONTROL REGISTER 2**

| Bit Range | Bit<br>31/23/15/7     | Bit<br>30/22/14/6     | Bit<br>29/21/13/5     | Bit<br>28/20/12/4     | Bit<br>27/19/11/3     | Bit<br>26/18/10/2     | Bit<br>25/17/9/1      | Bit<br>24/16/8/0      |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 31:24     | U-0                   | U-0                   | U-0                   | U-0                   | U-0                   | U-0                   | U-0                   | U-0                   |
|           | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     |
| 23:16     | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | DIFF27                | SIGN27                | DIFF26                | SIGN26                | DIFF25                | SIGN25                | DIFF24                | SIGN24                |
| 15:8      | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | DIFF23 <sup>(1)</sup> | SIGN23 <sup>(1)</sup> | DIFF22 <sup>(1)</sup> | SIGN22 <sup>(1)</sup> | DIFF21 <sup>(1)</sup> | SIGN21 <sup>(1)</sup> | DIFF20 <sup>(1)</sup> | SIGN20 <sup>(1)</sup> |
| 7:0       | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | DIFF19                | SIGN19                | DIFF18                | SIGN18                | DIFF17                | SIGN17                | DIFF16                | SIGN16                |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

- bit 31-24    **Unimplemented:** Read as '0'
- bit 23      **DIFF27:** AN27 Mode bit  
 1 = Selects AN27 differential pair input as AN27+ and AN1-  
 0 = AN27 is using Single-ended mode
- bit 22      **SIGN27:** AN27 Signed Data Mode bit  
 1 = AN27 is using Signed Data mode  
 0 = AN27 is using Unsigned Data mode
- bit 21      **DIFF26:** AN26 Mode bit  
 1 = Selects AN26 differential pair input as AN26+ and AN1-  
 0 = AN26 is using Single-ended mode
- bit 20      **SIGN26:** AN26 Signed Data Mode bit  
 1 = AN26 is using Signed Data mode  
 0 = AN26 is using Unsigned Data mode
- bit 19      **DIFF25:** AN25 Mode bit  
 1 = Selects AN25 differential pair input as AN25+ and AN1-  
 0 = AN25 is using Single-ended mode
- bit 18      **SIGN25:** AN25 Signed Data Mode bit  
 1 = AN25 is using Signed Data mode  
 0 = AN25 is using Unsigned Data mode
- bit 17      **DIFF24:** AN24 Mode bit  
 1 = Selects AN24 differential pair input as AN24+ and AN1-  
 0 = AN24 is using Single-ended mode
- bit 16      **SIGN24:** AN24 Signed Data Mode bit  
 1 = AN24 is using Signed Data mode  
 0 = AN24 is using Unsigned Data mode
- bit 15      **DIFF23:** AN23 Mode bit<sup>(1)</sup>  
 1 = Selects AN23 differential pair input as AN23+ and AN1-  
 0 = AN23 is using Single-ended mode
- bit 14      **SIGN23:** AN23 Signed Data Mode bit<sup>(1)</sup>  
 1 = AN23 is using Signed Data mode  
 0 = AN23 is using Unsigned Data mode

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-6: ADCIMCON2: ADC INPUT MODE CONTROL REGISTER 2 (CONTINUED)

|        |  |
|--------|--|
| bit 13 | <b>DIFF22:</b> AN22 Mode bit <sup>(1)</sup><br>1 = Selects AN22 differential pair input as AN22+ and AN1-<br>0 = AN22 is using Single-ended mode |
| bit 12 | <b>SIGN22:</b> AN22 Signed Data Mode bit <sup>(1)</sup><br>1 = AN22 is using Signed Data mode<br>0 = AN22 is using Unsigned Data mode            |
| bit 11 | <b>DIFF21:</b> AN21 Mode bit <sup>(1)</sup><br>1 = Selects AN21 differential pair input as AN21+ and AN1-<br>0 = AN21 is using Single-ended mode |
| bit 10 | <b>SIGN21:</b> AN21 Signed Data Mode bit <sup>(1)</sup><br>1 = AN21 is using Signed Data mode<br>0 = AN21 is using Unsigned Data mode            |
| bit 9  | <b>DIFF20:</b> AN20 Mode bit <sup>(1)</sup><br>1 = Selects AN20 differential pair input as AN20+ and AN1-<br>0 = AN20 is using Single-ended mode |
| bit 8  | <b>SIGN20:</b> AN20 Signed Data Mode bit <sup>(1)</sup><br>1 = AN20 is using Signed Data mode<br>0 = AN20 is using Unsigned Data mode            |
| bit 7  | <b>DIFF19:</b> AN19 Mode bit<br>1 = Selects AN19 differential pair input as AN19+ and AN1-<br>0 = AN19 is using Single-ended mode                |
| bit 6  | <b>SIGN19:</b> AN19 Signed Data Mode bit<br>1 = AN19 is using Signed Data mode<br>0 = AN19 is using Unsigned Data mode                           |
| bit 5  | <b>DIFF18:</b> AN18 Mode bit<br>1 = Selects AN18 differential pair input as AN18+ and AN1-<br>0 = AN18 is using Single-ended mode                |
| bit 4  | <b>SIGN18:</b> AN18 Signed Data Mode bit<br>1 = AN18 is using Signed Data mode<br>0 = AN18 is using Unsigned Data mode                           |
| bit 3  | <b>DIFF17:</b> AN17 Mode bit<br>1 = Selects AN17 differential pair input as AN17+ and AN1-<br>0 = AN17 is using Single-ended mode                |
| bit 2  | <b>SIGN17:</b> AN17 Signed Data Mode bit<br>1 = AN17 is using Signed Data mode<br>0 = AN17 is using Unsigned Data mode                           |
| bit 1  | <b>DIFF16:</b> AN16 Mode bit<br>1 = Selects AN16 differential pair input as AN16+ and AN1-<br>0 = AN16 is using Single-ended mode                |
| bit 0  | <b>SIGN16:</b> AN16 Signed Data Mode bit<br>1 = AN16 is using Signed Data mode<br>0 = AN16 is using Unsigned Data mode                           |

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-7: ADCIMCON3: ADC INPUT MODE CONTROL REGISTER 3**

| Bit Range | Bit<br>31/23/15/7     | Bit<br>30/22/14/6     | Bit<br>29/21/13/5     | Bit<br>28/20/12/4     | Bit<br>27/19/11/3     | Bit<br>26/18/10/2     | Bit<br>25/17/9/1      | Bit<br>24/16/8/0      |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 31:24     | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | U-0                   | U-0                   |
|           | DIFF47 <sup>(1)</sup> | SIGN47 <sup>(1)</sup> | DIFF46 <sup>(1)</sup> | SIGN46 <sup>(1)</sup> | DIFF45 <sup>(1)</sup> | SIGN45 <sup>(1)</sup> | —                     | —                     |
| 23:16     | U-0                   | U-0                   | U-0                   | U-0                   | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | —                     | —                     | —                     | —                     | DIFF41 <sup>(1)</sup> | SIGN41 <sup>(1)</sup> | DIFF40 <sup>(1)</sup> | SIGN40 <sup>(1)</sup> |
| 15:8      | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | DIFF39 <sup>(1)</sup> | SIGN39 <sup>(1)</sup> | DIFF38 <sup>(1)</sup> | SIGN38 <sup>(1)</sup> | DIFF37 <sup>(1)</sup> | SIGN37 <sup>(1)</sup> | DIFF36 <sup>(1)</sup> | SIGN36 <sup>(1)</sup> |
| 7:0       | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | U-0                   | U-0                   |
|           | DIFF35 <sup>(1)</sup> | SIGN35 <sup>(1)</sup> | DIFF34 <sup>(1)</sup> | SIGN34 <sup>(1)</sup> | DIFF33 <sup>(1)</sup> | SIGN33 <sup>(1)</sup> | —                     | —                     |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
-n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

- bit 31      **DIFF47:** AN47 Mode bit<sup>(1)</sup>  
1 = Selects AN47 differential input pair as AN47+ and AN1-  
0 = AN47 is using Single-ended mode
- bit 30      **SIGN47:** AN47 Signed Data Mode bit<sup>(1)</sup>  
1 = AN41 is using Signed Data mode  
0 = AN41 is using Unsigned Data mode
- bit 29      **DIFF46:** AN46 Mode bit<sup>(1)</sup>  
1 = Selects AN46 differential input pair as AN46+ and AN1-  
0 = AN41 is using Single-ended mode
- bit 28      **SIGN46:** AN46 Signed Data Mode bit<sup>(1)</sup>  
1 = AN46 is using Signed Data mode  
0 = AN46 is using Unsigned Data mode
- bit 27      **DIFF45:** AN45 Mode bit<sup>(1)</sup>  
1 = Selects AN45 differential input pair as AN45+ and AN1-  
0 = AN45 is using Single-ended mode
- bit 26      **SIGN46:** AN45 Signed Data Mode bit<sup>(1)</sup>  
1 = AN45 is using Signed Data mode  
0 = AN45 is using Unsigned Data mode
- bit 25-20      **Unimplemented:** Read as '0'
- bit 19      **DIFF41:** AN41 Mode bit<sup>(1)</sup>  
1 = Selects AN41 differential input pair as AN41+ and AN1-  
0 = AN41 is using Single-ended mode
- bit 18      **SIGN41:** AN41 Signed Data Mode bit<sup>(1)</sup>  
1 = AN41 is using Signed Data mode  
0 = AN41 is using Unsigned Data mode
- bit 17      **DIFF40:** AN40 Mode bit<sup>(1)</sup>  
1 = Selects AN40 differential input pair as AN40+ and AN1-  
0 = AN40 is using Single-ended mode
- bit 16      **SIGN40:** AN40 Signed Data Mode bit<sup>(1)</sup>  
1 = AN40 is using Signed Data mode  
0 = AN40 is using Unsigned Data mode

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-7: ADCIMCON3: ADC INPUT MODE CONTROL REGISTER 3 (CONTINUED)

|         |  |
|---------|--|
| bit 15  | <b>DIFF39:</b> AN39 Mode bit <sup>(1)</sup><br>1 = Selects AN39 differential input pair as AN39+ and AN1-<br>0 = AN39 is using Single-ended mode |
| bit 14  | <b>SIGN39:</b> AN39 Signed Data Mode bit <sup>(1)</sup><br>1 = AN39 is using Signed Data mode<br>0 = AN39 is using Unsigned Data mode            |
| bit 13  | <b>DIFF38:</b> AN38 Mode bit <sup>(1)</sup><br>1 = Selects AN38 differential input pair as AN38+ and AN1-<br>0 = AN38 is using Single-ended mode |
| bit 12  | <b>SIGN38:</b> AN38 Signed Data Mode bit <sup>(1)</sup><br>1 = AN38 is using Signed Data mode<br>0 = AN38 is using Unsigned Data mode            |
| bit 11  | <b>DIFF37:</b> AN37 Mode bit <sup>(1)</sup><br>1 = Selects AN37 differential input pair as AN37+ and AN1-<br>0 = AN37 is using Single-ended mode |
| bit 10  | <b>SIGN37:</b> AN37 Signed Data Mode bit <sup>(1)</sup><br>1 = AN37 is using Signed Data mode<br>0 = AN37 is using Unsigned Data mode            |
| bit 9   | <b>DIFF36:</b> AN36 Mode bit <sup>(1)</sup><br>1 = Selects AN36 differential input pair as AN36+ and AN1-<br>0 = AN36 is using Single-ended mode |
| bit 8   | <b>SIGN36:</b> AN36 Signed Data Mode bit <sup>(1)</sup><br>1 = AN36 is using Signed Data mode<br>0 = AN36 is using Unsigned Data mode            |
| bit 7   | <b>DIFF35:</b> AN35 Mode bit <sup>(1)</sup><br>1 = Selects AN35 differential input pair as AN35+ and AN1-<br>0 = AN35 is using Single-ended mode |
| bit 6   | <b>SIGN35:</b> AN35 Signed Data Mode bit <sup>(1)</sup><br>1 = AN35 is using Signed Data mode<br>0 = AN35 is using Unsigned Data mode            |
| bit 5   | <b>DIFF34:</b> AN34 Mode bit <sup>(1)</sup><br>1 = Selects AN34 differential input pair as AN34+ and AN1-<br>0 = AN34 is using Single-ended mode |
| bit 4   | <b>SIGN34:</b> AN34 Signed Data Mode bit <sup>(1)</sup><br>1 = AN34 is using Signed Data mode<br>0 = AN34 is using Unsigned Data mode            |
| bit 3   | <b>DIFF33:</b> AN33 Mode bit <sup>(1)</sup><br>1 = Selects AN33 differential input pair as AN33+ and AN1-<br>0 = AN33 is using Single-ended mode |
| bit 2   | <b>SIGN33:</b> AN33 Signed Data Mode bit <sup>(1)</sup><br>1 = AN33 is using Signed Data mode<br>0 = AN33 is using Unsigned Data mode            |
| bit 1-0 | <b>Unimplemented:</b> Read as '0'  |

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-8: ADCIMCON4: ADC INPUT MODE CONTROL REGISTER 4**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 7:0       | U-0               | U-0               | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | —                 | —                 | DIFF49            | SIGN49            | DIFF48           | SIGN48           |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

- bit 31-4      **Unimplemented:** Read as '0'
- bit 3      **DIFF49:** AN49 Mode bit
  - 1 = Selects AN49 differential input pair as AN49+ and AN1-
  - 0 = AN49 is using Single-ended mode
- bit 2      **SIGN49:** AN41 Signed Data Mode bit
  - 1 = AN49 is using Signed Data mode
  - 0 = AN49 is using Unsigned Data mode
- bit 1      **DIFF48:** AN48 Mode bit
  - 1 = Selects AN40 differential input pair as AN48+ and AN1-
  - 0 = AN48 is using Single-ended mode
- bit 0      **SIGN48:** AN48 Signed Data Mode bit
  - 1 = AN48 is using Signed Data mode
  - 0 = AN48 is using Unsigned Data mode

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-9: ADCGIRQEN1: ADC GLOBAL INTERRUPT ENABLE REGISTER 1

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6         | Bit 29/21/13/5         | Bit 28/20/12/4         | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|------------------------|------------------------|------------------------|------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                    | U-0                    | U-0                    | U-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —                      | —                      | —                      | —                      | AGIEN27        | AGIEN26        | AGIEN25       | AGIEN24       |
| 23:16     | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | AGIEN23 <sup>(1)</sup> | AGIEN22 <sup>(1)</sup> | AGIEN21 <sup>(1)</sup> | AGIEN20 <sup>(1)</sup> | AGIEN19        | AGIEN18        | AGIEN17       | AGIEN16       |
| 15:8      | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | AGIEN15                | AGIEN14                | AGIEN13                | AGIEN12                | AGIEN11        | AGIEN10        | AGIEN9        | AGIEN8        |
| 7:0       | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0                  | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | AGIEN7                 | AGIEN6                 | AGIEN5                 | AGIEN4                 | AGIEN3         | AGIEN2         | AGIEN1        | AGIEN0        |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **AGIEN27:AGIEN0:** ADC Global Interrupt Enable bits

1 = Interrupts are enabled for the selected analog input. The interrupt is generated after the converted data is ready (indicated by the AIRDYx bit of the ADCDSTAT1 register)

0 = Interrupts are disabled

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-10: ADCGIRQEN2: ADC GLOBAL INTERRUPT ENABLE REGISTER 2**

| Bit Range | Bit 31/23/15/7                  | Bit 30/22/14/6                  | Bit 29/21/13/5                  | Bit 28/20/12/4                  | Bit 27/19/11/3                  | Bit 26/18/10/2                  | Bit 25/17/9/1                   | Bit 24/16/8/0                   |
|-----------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| 31:24     | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        |
| 23:16     | U-0<br>—                        | U-0<br>—                        | R/W-0<br>AGIEN53                | U-0<br>—                        | U-0<br>—                        | R/W-0<br>AGIEN50                | R/W-0<br>AGIEN49                | R/W-0<br>AGIEN48                |
| 15:8      | R/W-0<br>AGIEN47 <sup>(1)</sup> | R/W-0<br>AGIEN46 <sup>(1)</sup> | R/W-0<br>AGIEN45 <sup>(1)</sup> | U-0<br>—                        | U-0<br>—                        | U-0<br>—                        | R/W-0<br>AGIEN41 <sup>(1)</sup> | R/W-0<br>AGIEN40 <sup>(1)</sup> |
| 7:0       | R/W-0<br>AGIEN39 <sup>(1)</sup> | R/W-0<br>AGIEN38 <sup>(1)</sup> | R/W-0<br>AGIEN37 <sup>(1)</sup> | R/W-0<br>AGIEN36 <sup>(1)</sup> | R/W-0<br>AGIEN35 <sup>(1)</sup> | R/W-0<br>AGIEN34 <sup>(1)</sup> | R/W-0<br>AGIEN33 <sup>(1)</sup> | U-0<br>—                        |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-22 **Unimplemented:** Read as '0'

bit 21 **AGIEN53:** ADC Global Interrupt Enable bit

1 = Interrupts are enabled for the selected analog input. The interrupt is generated after the converted data is ready (indicated by the AIRDYx bit of the ADCDSTAT2 register)  
 0 = Interrupts are disabled

bit 20-19 **Unimplemented:** Read as '0'

bit 18-13 **AGIEN50: AGIEN45:** ADC Global Interrupt Enable bits

1 = Interrupts are enabled for the selected analog input. The interrupt is generated after the converted data is ready (indicated by the AIRDYx bit of the ADCDSTAT2 register)  
 0 = Interrupts are disabled

bit 12-10 **Unimplemented:** Read as '0'

bit 9-1 **AGIEN41:AGIEN33** ADC Global Interrupt Enable bits

1 = Interrupts are enabled for the selected analog input. The interrupt is generated after the converted data is ready (indicated by the AIRDYx bit of the ADCDSTAT2 register)  
 0 = Interrupts are disabled

bit 0 **Unimplemented:** Read as '0'

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-11: ADCCSS1: ADC COMMON SCAN SELECT REGISTER 1

| Bit Range | Bit<br>31/23/15/7    | Bit<br>30/22/14/6    | Bit<br>29/21/13/5    | Bit<br>28/20/12/4    | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|----------------------|----------------------|----------------------|----------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0                  | U-0                  | U-0                  | U-0                  | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                    | —                    | —                    | —                    | CSS27             | CSS26             | CSS25            | CSS24            |
| 23:16     | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | CSS23 <sup>(1)</sup> | CSS22 <sup>(1)</sup> | CSS21 <sup>(1)</sup> | CSS20 <sup>(1)</sup> | CSS19             | CSS18             | CSS17            | CSS16            |
| 15:8      | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | CSS15                | CSS14                | CSS13                | CSS12                | CSS11             | CSS10             | CSS9             | CSS8             |
| 7:0       | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | CSS7                 | CSS6                 | CSS5                 | CSS4                 | CSS3              | CSS2              | CSS1             | CSS0             |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-28    **Unimplemented:** Read as '0'

bit 27-0    **CSS27:CSS0:** Analog Common Scan Select bits

Analog inputs AN27-AN6 are always Class 3 shared ADC7.

- 1 = Select ANx for input scan (i.e., ANx = CSSx and scan is sequential starting with the lowest to highest enabled CSSx analog input pin)
- 0 = Skip ANx for input scan

**Note 1:** This bit is not available on 64-pin devices.

**Note 1:** In addition to setting the appropriate bits in this register, Class 1 and Class 2 analog inputs must select the STRIG input as the trigger source if they are to be scanned through the CSSx bits. Refer to the bit descriptions in the ADCTRGx registers for selecting the STRIG option.

**2:** If a Class 1 or Class 2 input is included in the scan by setting the CSSx bit to '1' and by setting the TRGSRCx<4:0> bits to STRIG mode ( '0b11), the user application must ensure that no other triggers are generated for that input using the RQCNVRT bit in the ADCCON3 register or the hardware input or any digital filter. Otherwise, the scan behavior is unpredictable.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-12: ADCCSS2: ADC COMMON SCAN SELECT REGISTER 2

| Bit Range | Bit<br>31/23/15/7    | Bit<br>30/22/14/6    | Bit<br>29/21/13/5    | Bit<br>28/20/12/4    | Bit<br>27/19/11/3    | Bit<br>26/18/10/2    | Bit<br>25/17/9/1     | Bit<br>24/16/8/0     |
|-----------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| 31:24     | U-0                  | U-0                  | U-0                  | U-0                  | U-0                  | U-0                  | U-0                  | U-0                  |
|           | —                    | —                    | —                    | —                    | —                    | —                    | —                    | —                    |
| 23:16     | U-0                  | U-0                  | R/W-0                | U-0                  | U-0                  | R/W-0                | R/W-0                | R/W-0                |
|           | —                    | —                    | CSS53 <sup>(2)</sup> | —                    | —                    | CSS50 <sup>(2)</sup> | CSS49                | CSS48                |
| 15:8      | R/W-0                | R/W-0                | R/W-0                | U-0                  | U-0                  | U-0                  | R/W-0                | R/W-0                |
|           | CSS47 <sup>(1)</sup> | CSS46 <sup>(1)</sup> | CSS45 <sup>(1)</sup> | —                    | —                    | —                    | CSS41 <sup>(1)</sup> | CSS40 <sup>(1)</sup> |
| 7:0       | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0                | U-0                  |
|           | CSS39 <sup>(1)</sup> | CSS38 <sup>(1)</sup> | CSS37 <sup>(1)</sup> | CSS36 <sup>(1)</sup> | CSS35 <sup>(1)</sup> | CSS34 <sup>(1)</sup> | CSS33 <sup>(1)</sup> | —                    |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-22    **Unimplemented:** Read as '0'

bit 21        **CSS53:** Analog Common Scan Select bits  
               1 = Select ANx for input scan  
               0 = Skip ANx for input scan

bit 20-19    **Unimplemented:** Read as '0'

bit 18-13    **CSS50:CSS45:** Analog Common Scan Select bits

bit 12-10    **Unimplemented:** Read as '0'

bit 9-1        **CSS41:CSS33:** Analog Common Scan Select bits  
               1 = Select ANx for input scan  
               0 = Skip ANx for input scan

bit 0         **Unimplemented:** Read as '0'

**Note 1:** This bit is not available on 64-pin devices.

**2:** CSS50-CSS53 are internal analog inputs with respect to internal (IVREF, and CTMU) Temperature Sensor.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-13: ADCDSTAT1: ADC DATA READY STATUS REGISTER 1

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6         | Bit 29/21/13/5         | Bit 28/20/12/4         | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|------------------------|------------------------|------------------------|------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                    | U-0                    | U-0                    | U-0                    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | —                      | —                      | —                      | —                      | AIRDY27        | AIRDY26        | AIRDY25       | AIRDY24       |
| 23:16     | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | AIRDY23 <sup>(1)</sup> | AIRDY22 <sup>(1)</sup> | AIRDY21 <sup>(1)</sup> | AIRDY20 <sup>(1)</sup> | AIRDY19        | AIRDY18        | AIRDY17       | AIRDY16       |
| 15:8      | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | AIRDY15                | AIRDY14                | AIRDY13                | AIRDY12                | AIRDY11        | AIRDY10        | AIRDY9        | AIRDY8        |
| 7:0       | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | AIRDY7                 | AIRDY6                 | AIRDY5                 | AIRDY4                 | AIRDY3         | AIRDY2         | AIRDY1        | AIRDY0        |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **AIRDY27:AIRDY0:** Conversion Data Ready for Corresponding Analog Input Ready bits

1 = This bit is set when converted data is ready in the data register

0 = This bit is cleared when the associated data register is read

**Note 1:** This bit is not available on 64-pin devices.

## REGISTER 23-14: ADCDSTAT2: ADC DATA READY STATUS REGISTER 2

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6         | Bit 29/21/13/5         | Bit 28/20/12/4         | Bit 27/19/11/3         | Bit 26/18/10/2         | Bit 25/17/9/1          | Bit 24/16/8/0          |
|-----------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| 31:24     | U-0                    | U-0                    | U-0                    | U-0                    | U-0                    | U-0                    | U-0                    | U-0                    |
|           | —                      | —                      | —                      | —                      | —                      | —                      | —                      | —                      |
| 23:16     | U-0                    | U-0                    | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            |
|           | —                      | —                      | AIRDY53                | AIRDY52                | AIRDY51                | AIRDY50                | AIRDY49                | AIRDY48                |
| 15:8      | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | U-0                    | U-0                    | U-0                    | R-0, HS, HC            | R-0, HS, HC            |
|           | AIRDY47 <sup>(1)</sup> | AIRDY46 <sup>(1)</sup> | AIRDY45 <sup>(1)</sup> | —                      | —                      | —                      | AIRDY41 <sup>(1)</sup> | AIRDY40 <sup>(1)</sup> |
| 7:0       | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | U-0                    |
|           | AIRDY39 <sup>(1)</sup> | AIRDY38 <sup>(1)</sup> | AIRDY37 <sup>(1)</sup> | AIRDY36 <sup>(1)</sup> | AIRDY35 <sup>(1)</sup> | AIRDY34 <sup>(1)</sup> | AIRDY33 <sup>(1)</sup> | —                      |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-22 **Unimplemented:** Read as '0'

bit 23-13 **AIRDY53:AIRDY45:** Conversion Data Ready for Corresponding Analog Input Ready bits

1 = This bit is set when converted data is ready in the data register

0 = This bit is cleared when the associated data register is read

bit 12-10 **Unimplemented:** Read as '0'

bit 23-13 **AIRDY41:AIRDY33:** Conversion Data Ready for Corresponding Analog Input Ready bits

1 = This bit is set when converted data is ready in the data register

0 = This bit is cleared when the associated data register is read

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 23-15: ADCCMPENx: ADC DIGITAL COMPARATOR 'x' ENABLE REGISTER ('x' = 1 THROUGH 4)**

| Bit Range | Bit 31/23/15/7        | Bit 30/22/14/6        | Bit 29/21/13/5        | Bit 28/20/12/4        | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                   | U-0                   | U-0                   | U-0                   | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —                     | —                     | —                     | —                     | CMPE27         | CMPE26         | CMPE25        | CMPE24        |
| 23:16     | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPE23 <sup>(1)</sup> | CMPE22 <sup>(1)</sup> | CMPE21 <sup>(1)</sup> | CMPE20 <sup>(1)</sup> | CMPE19         | CMPE18         | CMPE17        | CMPE16        |
| 15:8      | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPE15                | CMPE14                | CMPE13                | CMPE12                | CMPE11         | CMPE10         | CMPE9         | CMPE8         |
| 7:0       | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPE7                 | CMPE6                 | CMPE5                 | CMPE4                 | CMPE3          | CMPE2          | CMPE1         | CMPE0         |

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **CMPE27:CMPE0:** ADC Digital Comparator 'x' Enable bits

These bits enable conversion results corresponding to the Analog Input to be processed by the Digital Comparator. CMPE0 enables AN0, CMPE1 enables AN1, and so on.

**Note 1:** This bit is not available on 64-pin devices.

**Note 1:** CMPE<sub>x</sub> = AN<sub>x</sub>, where 'x' = 0-31 (Digital Comparator inputs are limited to AN0 through AN31).  
**2:** Changing the bits in this register while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-16: ADCCMPx: ADC DIGITAL COMPARATOR 'x' LIMIT VALUE REGISTER (‘x’ = 1 THROUGH 4)

| Bit Range | Bit<br>31/23/15/7               | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|---------------------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R/W-0                           | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DCMPHI<15:8> <sup>(1,2,3)</sup> |                   |                   |                   |                   |                   |                  |                  |
| 23:16     | R/W-0                           | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DCMPHI<7:0> <sup>(1,2,3)</sup>  |                   |                   |                   |                   |                   |                  |                  |
| 15:8      | R/W-0                           | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DCMPLO<15:8> <sup>(1,2,3)</sup> |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0                           | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DCMPLO<7:0> <sup>(1,2,3)</sup>  |                   |                   |                   |                   |                   |                  |                  |

### Legend:

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

- bit 31-16 **DCMPHI<15:0>**: Digital Comparator 'x' High Limit Value bits<sup>(1,2,3)</sup>  
 These bits store the high limit value, which is used by digital comparator for comparisons with ADC converted data.
- bit 15-0 **DCMPLO<15:0>**: Digital Comparator 'x' Low Limit Value bits<sup>(1,2,3)</sup>  
 These bits store the low limit value, which is used by digital comparator for comparisons with ADC converted data.

- Note 1:** Changing these bits while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.
- 2:** The format of the limit values should match the format of the ADC converted value in terms of sign and fractional settings.
- 3:** For Digital Comparator 0 used in CVD mode, the DCMPHI<15:0> and DCMPLO<15:0> bits must always be specified in signed format, as the CVD output data is differential and is always signed.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-17: ADCFLTRx: ADC DIGITAL FILTER 'x' REGISTER (‘x’ = 1 THROUGH 6)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R-0, HS, HC   |
|           | AFEN           | DATA16EN       | DFMODE         | OVRSAM<2:0>    |                |                | AFGIEN        | AFRDY         |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | CHNLID<4:0>    |                |                |               |               |
| 15:8      | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | FLTRDATA<15:8> |                |                |                |                |                |               |               |
| 7:0       | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | FLTRDATA<7:0>  |                |                |                |                |                |               |               |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as ‘0’           |
| -n = Value at POR | ‘1’ = Bit is set  | ‘0’ = Bit is cleared      x = Bit is unknown |

- bit 31 **AFEN:** Digital Filter ‘x’ Enable bit  
 1 = Digital filter is enabled  
 0 = Digital filter is disabled and the AFRDY status bit is cleared  
**NOTE:** If ADCFLTRx<AFEN> = 1 then ADCxTIME<SAMC> minimum must be ≥ to 0x004 or equivalent to ≥ 6 TAD. This will correspondingly reduce the maximum sampling rate.
- bit 30 **DATA16EN:** Filter Significant Data Length bit  
 1 = All 16 bits of the filter output data are significant  
 0 = Only the first 12 bits are significant, followed by four zeros  
**Note:** This bit is significant only if DFMODE = 1 (Averaging Mode) and FRACT (ADCCON1<23>) = 1 (Fractional Output Mode).
- bit 29 **DFMODE:** ADC Filter Mode bit  
 1 = Filter ‘x’ works in Averaging mode  
 0 = Filter ‘x’ works in Oversampling Filter mode (default)
- bit 28-26 **OVRSAM<2:0>:** Oversampling Filter Ratio bits  
**If DFMODE is ‘0’:**  
 111 = 128 samples (shift sum 3 bits to right, output data is in 15.1 format)  
 110 = 32 samples (shift sum 2 bits to right, output data is in 14.1 format)  
 101 = 8 samples (shift sum 1 bit to right, output data is in 13.1 format)  
 100 = 2 samples (shift sum 0 bits to right, output data is in 12.1 format)  
 011 = 256 samples (shift sum 4 bits to right, output data is 16 bits)  
 010 = 64 samples (shift sum 3 bits to right, output data is 15 bits)  
 001 = 16 samples (shift sum 2 bits to right, output data is 14 bits)  
 000 = 4 samples (shift sum 1 bit to right, output data is 13 bits)
- If DFMODE is ‘1’:**  
 111 = 256 samples (256 samples to be averaged)  
 110 = 128 samples (128 samples to be averaged)  
 101 = 64 samples (64 samples to be averaged)  
 100 = 32 samples (32 samples to be averaged)  
 011 = 16 samples (16 samples to be averaged)  
 010 = 8 samples (8 samples to be averaged)  
 001 = 4 samples (4 samples to be averaged)  
 000 = 2 samples (2 samples to be averaged)

**Note 1:** This selection is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-17: ADCFLTRx: ADC DIGITAL FILTER 'x' REGISTER (‘x’ = 1 THROUGH 6) (CONTINUED)

bit 25 **AFGIEN:** Digital Filter 'x' Interrupt Enable bit  
1 = Digital filter interrupt is enabled and is generated by the AFRDY status bit  
0 = Digital filter is disabled

bit 24 **AFRDY:** Digital Filter 'x' Data Ready Status bit  
1 = Data is ready in the FLTRDATA<15:0> bits  
0 = Data is not ready

**Note:** This bit is cleared by reading the FLTRDATA<15:0> bits or by disabling the Digital Filter module (by setting AFEN to '0').

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **CHNLID<4:0>:** Digital Filter Analog Input Selection bits  
These bits specify the analog input to be used as the oversampling filter data source.

11111 = Reserved  
.  
.  
.  
11100 = Reserved  
11011 = AN27 input  
11010 = AN26 input  
11001 = AN25 input  
11000 = AN24 input  
10111 = AN23<sup>(1)</sup> input  
10110 = AN22<sup>(1)</sup> input  
10101 = AN21<sup>(1)</sup> input  
10100 = AN20<sup>(1)</sup> input  
10011 = AN19 input  
.  
.  
10110 = AN6 input  
00101 = ADC5 Module  
00100 = ADC4 Module  
00011 = ADC3 Module  
00010 = ADC2 Module  
00001 = ADC1 Module  
00000 = ADC0 Module

**Note:** Only the first 32 analog inputs (Class 1 and Class 2) can use a digital filter.

bit 15-0 **FLTRDATA<15:0>:** Digital Filter 'x' Data Output Value bits  
The filter output data is as per the fractional format set in the FRACT (ADCCON1<23>) bit. The FRACT bit should not be changed while the filter is enabled. Changing the state of the FRACT bit after the operation of the filter ended will not update the value of FLTRDATA<15:0> to reflect the new format.

**Note 1:** This selection is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-18: ADCTRG1: ADC TRIGGER SOURCE 1 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC3<4:0>   |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC2<4:0>   |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC1<4:0>   |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC0<4:0>   |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC3<4:0>**: Trigger Source for Conversion of ADC3 Module Select bits

11111 = Reserved  
 11110 = PWM Generator 7 Current-Limit<sup>(1)</sup>  
 11101 = PWM Generator 6 Current-Limit<sup>(1)</sup>  
 11100 = PWM Generator 5 Current-Limit<sup>(1)</sup>  
 11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
 11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
 11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
 11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
 10111 = PWM Generator 9 trigger<sup>(1)</sup>  
 10110 = PWM Generator 8 trigger<sup>(1)</sup>  
 10101 = PWM Generator 7 trigger<sup>(1)</sup>  
 10100 = Reserved  
 10011 = Output Compare 4 rising edge  
 10010 = Output Compare 3 rising edge  
 10001 = Output Compare 2 rising edge  
 10000 = Output Compare 1 rising edge  
 01111 = PWM Generator 6 trigger<sup>(1)</sup>  
 01110 = PWM Generator 5 trigger<sup>(1)</sup>  
 01101 = PWM Generator 4 trigger<sup>(1)</sup>  
 01100 = PWM Generator 3 trigger<sup>(1)</sup>  
 01011 = PWM Generator 2 trigger<sup>(1)</sup>  
 01010 = PWM Generator 1 trigger<sup>(1)</sup>  
 01001 = Secondary Special Event Trigger<sup>(1)</sup>  
 01000 = Primary Special Event Trigger<sup>(1)</sup>  
 00111 = General Purpose Timer5  
 00110 = General Purpose Timer3  
 00101 = General Purpose Timer1  
 00100 = INT0  
 00011 = Scan trigger<sup>(2)</sup>  
 00010 = Software level trigger (must also configure the ADCTRGNS register)  
 00001 = Software edge trigger  
 00000 = No Trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXXMCXX Motor Control devices.

**Note 2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-18: ADCTRG1: ADC TRIGGER SOURCE 1 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC2<4:0>:** Trigger Source for Conversion of ADC2 Module Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC1<4:0>:** Trigger Source for Conversion of ADC1 Module Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC0<4:0>:** Trigger Source for Conversion of ADC0 Module Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-19: ADCTRG2: ADC TRIGGER SOURCE 2 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC7<4:0>   |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC6<4:0>   |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC5<4:0>   |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC4<4:0>   |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
-n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC7<4:0>:** Trigger Source for Conversion of Analog Input AN7 Select bits

11111 = Reserved  
11110 = PWM Generator 12 trigger<sup>(1)</sup>  
11101 = PWM Generator 11 trigger<sup>(1)</sup>  
11100 = PWM Generator 10 trigger<sup>(1)</sup>  
11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
10111 = PWM Generator 9 trigger<sup>(1)</sup>  
10110 = PWM Generator 8 trigger<sup>(1)</sup>  
10101 = PWM Generator 7 trigger<sup>(1)</sup>  
10100 = CTMU trip  
10011 = Output Compare 4 rising edge  
10010 = Output Compare 3 rising edge  
10001 = Output Compare 2 rising edge  
10000 = Output Compare 1 rising edge  
01111 = PWM Generator 6 trigger<sup>(1)</sup>  
01110 = PWM Generator 5 trigger<sup>(1)</sup>  
01101 = PWM Generator 4 trigger<sup>(1)</sup>  
01100 = PWM Generator 3 trigger<sup>(1)</sup>  
01011 = PWM Generator 2 trigger<sup>(1)</sup>  
01010 = PWM Generator 1 trigger<sup>(1)</sup>  
01001 = Secondary Special Event Trigger<sup>(1)</sup>  
01000 = Primary Special Event Trigger<sup>(1)</sup>  
00111 = General Purpose Timer5  
00110 = General Purpose Timer3  
00101 = General Purpose Timer1  
00100 = INT0  
00011 = Scan trigger<sup>(2)</sup>  
00010 = Software level trigger (must also configure the ADCTRGNS register)  
00001 = Software edge trigger  
00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-19: ADCTRG2: ADC TRIGGER SOURCE 2 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC6<4:0>:** Trigger Source for Conversion of Analog Input AN6 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC5<4:0>:** Trigger Source for Conversion of ADC5 Module Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC4<4:0>:** Trigger Source for Conversion of ADC4 Module Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-20: ADCTRG3: ADC TRIGGER SOURCE 3 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC11<4:0>  |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC10<4:0>  |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC9<4:0>   |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC8<4:0>   |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC11<4:0>**: Trigger Source for Conversion of Analog Input AN11 Select bits

11111 = Reserved  
 11110 = PWM Generator 12 trigger<sup>(1)</sup>  
 11101 = PWM Generator 11 trigger<sup>(1)</sup>  
 11100 = PWM Generator 10 trigger<sup>(1)</sup>  
 11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
 11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
 11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
 11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
 10111 = PWM Generator 9 trigger<sup>(1)</sup>  
 10110 = PWM Generator 8 trigger<sup>(1)</sup>  
 10101 = PWM Generator 7 trigger<sup>(1)</sup>  
 10100 = CTMU trip  
 10011 = Output Compare 4 rising edge  
 10010 = Output Compare 3 rising edge  
 10001 = Output Compare 2 rising edge  
 10000 = Output Compare 1 rising edge  
 01111 = PWM Generator 6 trigger<sup>(1)</sup>  
 01110 = PWM Generator 5 trigger<sup>(1)</sup>  
 01101 = PWM Generator 4 trigger<sup>(1)</sup>  
 01100 = PWM Generator 3 trigger<sup>(1)</sup>  
 01011 = PWM Generator 2 trigger<sup>(1)</sup>  
 01010 = PWM Generator 1 trigger<sup>(1)</sup>  
 01001 = Secondary Special Event Trigger<sup>(1)</sup>  
 01000 = Primary Special Event Trigger<sup>(1)</sup>  
 00111 = General Purpose Timer5  
 00110 = General Purpose Timer3  
 00101 = General Purpose Timer1  
 00100 = INT0  
 00011 = Scan trigger<sup>(2)</sup>  
 00010 = Software level trigger (must also configure the ADCTRGNS register)  
 00001 = Software edge trigger  
 00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-20: ADCTRG3: ADC TRIGGER SOURCE 3 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC10<4:0>:** Trigger Source for Conversion of Analog Input AN10 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC9<4:0>:** Trigger Source for Conversion of Analog Input AN9 Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC8<4:0>:** Trigger Source for Conversion of Analog Input AN8 Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-21: ADCTRG4: ADC TRIGGER SOURCE 4 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC15<4:0>  |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC14<4:0>  |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC13<4:0>  |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC12<4:0>  |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
-n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC15<4:0>**: Trigger Source for Conversion of Analog Input AN15 Select bits

11111 = Reserved  
11110 = PWM Generator 12 trigger<sup>(1)</sup>  
11101 = PWM Generator 11 trigger<sup>(1)</sup>  
11100 = PWM Generator 10 trigger<sup>(1)</sup>  
11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
10111 = PWM Generator 9 trigger<sup>(1)</sup>  
10110 = PWM Generator 8 trigger<sup>(1)</sup>  
10101 = PWM Generator 7 trigger<sup>(1)</sup>  
10100 = CTMU trip  
10011 = Output Compare 4 rising edge  
10010 = Output Compare 3 rising edge  
10001 = Output Compare 2 rising edge  
10000 = Output Compare 1 rising edge  
01111 = PWM Generator 6 trigger<sup>(1)</sup>  
01110 = PWM Generator 5 trigger<sup>(1)</sup>  
01101 = PWM Generator 4 trigger<sup>(1)</sup>  
01100 = PWM Generator 3 trigger<sup>(1)</sup>  
01011 = PWM Generator 2 trigger<sup>(1)</sup>  
01010 = PWM Generator 1 trigger<sup>(1)</sup>  
01001 = Secondary Special Event Trigger<sup>(1)</sup>  
01000 = Primary Special Event Trigger<sup>(1)</sup>  
00111 = General Purpose Timer5  
00110 = General Purpose Timer3  
00101 = General Purpose Timer1  
00100 = INT0  
00011 = Scan trigger<sup>(2)</sup>  
00010 = Software level trigger (must also configure the ADCTRGNS register)  
00001 = Software edge trigger  
00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-21: ADCTRG4: ADC TRIGGER SOURCE 4 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC14<4:0>**: Trigger Source for Conversion of Analog Input AN14 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC13<4:0>**: Trigger Source for Conversion of Analog Input AN13 Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC12<4:0>**: Trigger Source for Conversion of Analog Input AN12 Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-22: ADCTRG5: ADC TRIGGER SOURCE 5 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4               | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|------------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0                        | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC19<4:0> <sup>(3)</sup> |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0                        | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC18<4:0>                |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0                        | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC17<4:0>                |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0                        | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC16<4:0>                |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
-n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC19<4:0>**: Trigger Source for Conversion of Analog Input AN19 Select bits<sup>(3)</sup>

11111 = Reserved  
11110 = PWM Generator 12 trigger<sup>(1)</sup>  
11101 = PWM Generator 11 trigger<sup>(1)</sup>  
11100 = PWM Generator 10 trigger<sup>(1)</sup>  
11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
10111 = PWM Generator 9 trigger<sup>(1)</sup>  
10110 = PWM Generator 8 trigger<sup>(1)</sup>  
10101 = PWM Generator 7 trigger<sup>(1)</sup>  
10100 = CTMU trip  
10011 = Output Compare 4 rising edge  
10010 = Output Compare 3 rising edge  
10001 = Output Compare 2 rising edge  
10000 = Output Compare 1 rising edge  
01111 = PWM Generator 6 trigger<sup>(1)</sup>  
01110 = PWM Generator 5 trigger<sup>(1)</sup>  
01101 = PWM Generator 4 trigger<sup>(1)</sup>  
01100 = PWM Generator 3 trigger<sup>(1)</sup>  
01011 = PWM Generator 2 trigger<sup>(1)</sup>  
01010 = PWM Generator 1 trigger<sup>(1)</sup>  
01001 = Secondary Special Event Trigger<sup>(1)</sup>  
01000 = Primary Special Event Trigger<sup>(1)</sup>  
00111 = General Purpose Timer5  
00110 = General Purpose Timer3  
00101 = General Purpose Timer1  
00100 = INT0  
00011 = Scan trigger<sup>(2)</sup>  
00010 = Software level trigger (must also configure the ADCTRGNS register)  
00001 = Software edge trigger  
00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

**3:** These bits are not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-22: ADCTRG5: ADC TRIGGER SOURCE 5 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC18<4:0>**: Trigger Source for Conversion of Analog Input AN18 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC17<4:0>**: Trigger Source for Conversion of Analog Input AN17 Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC16<4:0>**: Trigger Source for Conversion of Analog Input AN16 Select bits  
See bits 28-24 for bit value definitions.

- Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.
- 2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.
- 3:** These bits are not available on 64-pin devices.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-23: ADCTRG6: ADC TRIGGER SOURCE 6 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC23<4:0>  |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC22<4:0>  |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC21<4:0>  |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC20<4:0>  |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC23<4:0>**: Trigger Source for Conversion of Analog Input AN23 Select bits

11111 = Reserved  
 11110 = PWM Generator 12 trigger<sup>(1)</sup>  
 11101 = PWM Generator 11 trigger<sup>(1)</sup>  
 11100 = PWM Generator 10 trigger<sup>(1)</sup>  
 11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
 11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
 11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
 11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
 10111 = PWM Generator 9 trigger<sup>(1)</sup>  
 10110 = PWM Generator 8 trigger<sup>(1)</sup>  
 10101 = PWM Generator 7 trigger<sup>(1)</sup>  
 10100 = CTMU trip  
 10011 = Output Compare 4 rising edge  
 10010 = Output Compare 3 rising edge  
 10001 = Output Compare 2 rising edge  
 10000 = Output Compare 1 rising edge  
 01111 = PWM Generator 6 trigger<sup>(1)</sup>  
 01110 = PWM Generator 5 trigger<sup>(1)</sup>  
 01101 = PWM Generator 4 trigger<sup>(1)</sup>  
 01100 = PWM Generator 3 trigger<sup>(1)</sup>  
 01011 = PWM Generator 2 trigger<sup>(1)</sup>  
 01010 = PWM Generator 1 trigger<sup>(1)</sup>  
 01001 = Secondary Special Event Trigger<sup>(1)</sup>  
 01000 = Primary Special Event Trigger<sup>(1)</sup>  
 00111 = General Purpose Timer5  
 00110 = General Purpose Timer3  
 00101 = General Purpose Timer1  
 00100 = INT0  
 00011 = Scan trigger<sup>(2)</sup>  
 00010 = Software level trigger (must also configure the ADCTRGNS register)  
 00001 = Software edge trigger  
 00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

**Note:** This register is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-23: ADCTRG6: ADC TRIGGER SOURCE 6 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC22<4:0>**: Trigger Source for Conversion of Analog Input AN22 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC21<4:0>**: Trigger Source for Conversion of Analog Input AN21 Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC20<4:0>**: Trigger Source for Conversion of Analog Input AN20 Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

**Note:** This register is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-24: ADCTR7: ADC TRIGGER SOURCE 7 REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC27<4:0>  |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC26<4:0>  |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC25<4:0>  |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | TRGSRC24<4:0>  |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
-n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **TRGSRC27<4:0>**: Trigger Source for Conversion of Analog Input AN27 Select bits

11111 = Reserved  
11110 = PWM Generator 12 trigger<sup>(1)</sup>  
11101 = PWM Generator 11 trigger<sup>(1)</sup>  
11100 = PWM Generator 10 trigger<sup>(1)</sup>  
11011 = PWM Generator 4 Current-Limit<sup>(1)</sup>  
11010 = PWM Generator 3 Current-Limit<sup>(1)</sup>  
11001 = PWM Generator 2 Current-Limit<sup>(1)</sup>  
11000 = PWM Generator 1 Current-Limit<sup>(1)</sup>  
10111 = PWM Generator 9 trigger<sup>(1)</sup>  
10110 = PWM Generator 8 trigger<sup>(1)</sup>  
10101 = PWM Generator 7 trigger<sup>(1)</sup>  
10100 = CTMU trip  
10011 = Output Compare 4 rising edge  
10010 = Output Compare 3 rising edge  
10001 = Output Compare 2 rising edge  
10000 = Output Compare 1 rising edge  
01111 = PWM Generator 6 trigger<sup>(1)</sup>  
01110 = PWM Generator 5 trigger<sup>(1)</sup>  
01101 = PWM Generator 4 trigger<sup>(1)</sup>  
01100 = PWM Generator 3 trigger<sup>(1)</sup>  
01011 = PWM Generator 2 trigger<sup>(1)</sup>  
01010 = PWM Generator 1 trigger<sup>(1)</sup>  
01001 = Secondary Special Event Trigger<sup>(1)</sup>  
01000 = Primary Special Event Trigger<sup>(1)</sup>  
00111 = General Purpose Timer5  
00110 = General Purpose Timer3  
00101 = General Purpose Timer1  
00100 = INT0  
00011 = Scan trigger<sup>(2)</sup>  
00010 = Software level trigger (must also configure the ADCTRNS register)  
00001 = Software edge trigger  
00000 = No trigger

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

**Note:** This register is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-24: ADCTRG7: ADC TRIGGER SOURCE 7 REGISTER

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **TRGSRC26<4:0>**: Trigger Source for Conversion of Analog Input AN26 Select bits  
See bits 28-24 for bit value definitions.

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC25<4:0>**: Trigger Source for Conversion of Analog Input AN25 Select bits  
See bits 28-24 for bit value definitions.

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TRGSRC24<4:0>**: Trigger Source for Conversion of Analog Input AN24 Select bits  
See bits 28-24 for bit value definitions.

**Note 1:** The PWM bit definitions are only applicable to PIC32MKXXMCXX Motor Control devices.

**2:** For Scan Trigger, in addition to setting the trigger, it also requires programming of the STRGSRC<4:0> bits (ADCCON1<20:16>) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

**Note:** This register is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-25: ADCCMPCON1: ADC DIGITAL COMPARATOR 1 CONTROL REGISTER

| Bit Range  | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |  |        |          |        |        |        |        |        |        |
|--|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|--|--------|----------|--------|--------|--------|--------|--------|--------|
| 31:24  | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |  |        |          |        |        |        |        |        |        |
| CVDDATA<15:8>  |                |                |                |                |                |                |               |               |  |        |          |        |        |        |        |        |        |
| 23:16  | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |  |        |          |        |        |        |        |        |        |
| CVDDATA<7:0>   |                |                |                |                |                |                |               |               |  |        |          |        |        |        |        |        |        |
| 15:8   | U-0            | U-0            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |  |        |          |        |        |        |        |        |        |
| AINID<5:0>   |                |                |                |                |                |                |               |               |  |        |          |        |        |        |        |        |        |
| 7:0  | R/W-0          | R/W-0          | R-0, HS, HC    | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |  |        |          |        |        |        |        |        |        |
| <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;">ENDCMP</td> <td style="width: 12.5%;">DCMPGIEN</td> <td style="width: 12.5%;">DCMPED</td> <td style="width: 12.5%;">IEBTWN</td> <td style="width: 12.5%;">IEHIHI</td> <td style="width: 12.5%;">IEHILO</td> <td style="width: 12.5%;">IELOHI</td> <td style="width: 12.5%;">IELOLO</td> </tr> </table> |                |                |                |                |                |                |               |               |  | ENDCMP | DCMPGIEN | DCMPED | IEBTWN | IEHIHI | IEHILO | IELOHI | IELOLO |
|  | ENDCMP         | DCMPGIEN       | DCMPED         | IEBTWN         | IEHIHI         | IEHILO         | IELOHI        | IELOLO        |  |        |          |        |        |        |        |        |        |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **CVDDATA<15:0>**: CVD Data Status bits

In CVD mode, these bits obtain the CVD differential output data (subtraction of CVD positive and negative measurement), whenever a Digital Comparator interrupt is generated. The value in these bits is compliant with the FRACT bit (ADCCON1<23>) and is always signed.

bit 15-14 **Unimplemented**: Read as '0'

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-25: ADCCMPCON1: ADC DIGITAL COMPARATOR 1 CONTROL REGISTER (CONTINUED)

bit 13-8 **AINID<5:0>**: Digital Comparator 1 Analog Input Identification (ID) bits  
When a digital comparator event occurs (DCMPED = 1), these bits identify the analog input being monitored by Digital Comparator 1.

**Note:** In normal ADC mode, only analog inputs <31:0> can be processed by the Digital Comparator 1. The Digital Comparator 1 also supports the CVD mode, in which all Class 2 and Class 3 analog inputs may be stored in the AINID<5:0> bits.

111111 = Reserved  
.  
.  
.  
110110 = Reserved  
110101 = Internal AN53 (CTMU temperature sensor)  
110101 = Reserved  
110101 = Reserved  
110010 = Internal AN50 (IVREF 1.2V)  
110001 = AN49 is being monitored  
.  
.  
.  
101101 = AN45 is being monitored  
101100 = Reserved  
.  
.  
.  
101010 = Reserved  
101001 = AN41 is being monitored  
.  
.  
.  
100001 = AN33 is being monitored  
111100 = Reserved  
.  
.  
.  
111000 = Reserved  
111011 = AN27 is being monitored  
.  
.  
.  
000000 = AN0 is being monitored

**Note:** For 64 pin devices AN20-AN23 and AN33-AN47 inputs above are not implemented.

bit 7 **ENDCMP**: Digital Comparator 1 Enable bit  
1 = Digital Comparator 1 is enabled  
0 = Digital Comparator 1 is not enabled, and the DCMPED status bit (ADCCMP0CON<5>) is cleared

bit 6 **DCMPGIEN**: Digital Comparator 1 Global Interrupt Enable bit  
1 = A Digital Comparator 1 interrupt is generated when the DCMPED status bit (ADCCMP0CON<5>) is set  
0 = A Digital Comparator 1 interrupt is disabled

bit 5 **DCMPED**: Digital Comparator 1 “Output True” Event Status bit  
The logical conditions under which the digital comparator gets “True” are defined by the IEBTWN, IEHIHI, IEHILO, IELOHI, and IELOLO bits.

**Note:** This bit is cleared by reading the AINID<5:0> bits or by disabling the Digital Comparator module (by setting ENDCMP to ‘0’).

1 = Digital Comparator 1 output true event has occurred (output of Comparator is ‘1’)  
0 = Digital Comparator 1 output is false (output of comparator is ‘0’)

bit 4 **IEBTWN**: Between Low/High Digital Comparator 1 Event bit  
1 = Generate a digital comparator event when  $DCMPLO<15:0> \leq DATA<31:0> < DCMPHI<15:0>$   
0 = Do not generate a digital comparator event

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-25: ADCCMPCON1: ADC DIGITAL COMPARATOR 1 CONTROL REGISTER (CONTINUED)

- bit 3     **IEHIHI:** High/High Digital Comparator 0 Event bit  
1 = Generate a Digital Comparator 0 Event when  $DCMPHI<15:0> \leq DATA<31:0>$   
0 = Do not generate an event
- bit 2     **IEHILO:** High/Low Digital Comparator 0 Event bit  
1 = Generate a Digital Comparator 0 Event when  $DATA<31:0> < DCMPHI<15:0>$   
0 = Do not generate an event
- bit 1     **IELOHI:** Low/High Digital Comparator 0 Event bit  
1 = Generate a Digital Comparator 0 Event when  $DCMPLO<15:0> \leq DATA<31:0>$   
0 = Do not generate an event
- bit 0     **IELOLO:** Low/Low Digital Comparator 0 Event bit  
1 = Generate a Digital Comparator 0 Event when  $DATA<31:0> < DCMPLO<15:0>$   
0 = Do not generate an event

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-26: ADCCMPCONx: ADC DIGITAL COMPARATOR 'x' CONTROL REGISTER ( 'x' = 2 THROUGH 4 )

| Bit Range | Bit 31/23/15/7  | Bit 30/22/14/6    | Bit 29/21/13/5        | Bit 28/20/12/4  | Bit 27/19/11/3  | Bit 26/18/10/2  | Bit 25/17/9/1   | Bit 24/16/8/0   |
|-----------|-----------------|-------------------|-----------------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| 31:24     | U-0<br>—        | U-0<br>—          | U-0<br>—              | U-0<br>—        | U-0<br>—        | U-0<br>—        | U-0<br>—        | U-0<br>—        |
| 23:16     | U-0<br>—        | U-0<br>—          | U-0<br>—              | U-0<br>—        | U-0<br>—        | U-0<br>—        | U-0<br>—        | U-0<br>—        |
| 15:8      | U-0<br>—        | U-0<br>—          | U-0<br>—              | R-0, HS, HC     | R-0, HS, HC     | R-0, HS, HC     | R-0, HS, HC     | R-0, HS, HC     |
|           | AINID<4:0>      |                   |                       |                 |                 |                 |                 |                 |
| 7:0       | R/W-0<br>ENDCMP | R/W-0<br>DCMPGIEN | R-0, HS, HC<br>DCMPED | R/W-0<br>IEBTWN | R/W-0<br>IEHIHI | R/W-0<br>IEHILO | R/W-0<br>IELOHI | R/W-0<br>IELOLO |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Cleared                        |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-13 **Unimplemented:** Read as '0'

bit 12-8 **AINID<4:0>:** Digital Comparator 'x' Analog Input Identification (ID) bits

When a digital comparator event occurs (DCMPED = 1), these bits identify the analog input being monitored by the Digital Comparator.

**Note:** Only analog inputs <27:0> can be processed by the Digital Comparator module 'x' ('x' = 2-4).

11111 = Reserved  
 .  
 .  
 11100 = Reserved  
 11011 = AN27  
 11010 = AN26  
 11001 = AN25  
 11000 = AN24  
 10111 = AN23<sup>(1)</sup>  
 10110 = AN22<sup>(1)</sup>  
 10101 = AN21<sup>(1)</sup>  
 10100 = AN20<sup>(1)</sup>  
 10011 = AN19  
 .  
 .  
 00001 = AN1  
 00000 = AN0

bit 7 **ENDCMP:** Digital Comparator 'x' Enable bit

1 = Digital Comparator 'x' is enabled

0 = Digital Comparator 'x' is not enabled, and the DCMPED status bit (ADCCMPxCON<5>) is cleared

bit 6 **DCMPGIEN:** Digital Comparator 'x' Global Interrupt Enable bit

1 = A Digital Comparator 'x' interrupt is generated when the DCMPED status bit (ADCCMPxCON<5>) is set

0 = A Digital Comparator 'x' interrupt is disabled

**Note 1:** This setting is not available on 64-pin devices.



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-26: ADCCMPCONx: ADC DIGITAL COMPARATOR 'x' CONTROL REGISTER ('x' = 2 THROUGH 4) (CONTINUED)

- bit 5 **DCMPED**: Digital Comparator 'x' "Output True" Event Status bit  
The logical conditions under which the digital comparator gets "True" are defined by the IEBTWN, IEHIHI, IEHILO, IELOHI and IELOLO bits.  
This bit is cleared by reading the AINID<5:0> bits (ADCCMPCONx<13:8>) or by disabling the Digital Comparator module (by setting ENDCMP to '0').  
1 = Digital Comparator 'x' output true event has occurred (output of Comparator is '1')  
0 = Digital Comparator 'x' output is false (output of Comparator is '0')
- bit 4 **IEBTWN**: Between Low/High Digital Comparator 'x' Event bit  
1 = Generate a digital comparator event when the DCMPL0<15:0> bits ≤ DATA<31:0> bits < DCMPHI<15:0> bits  
0 = Do not generate a digital comparator event
- bit 3 **IEHIHI**: High/High Digital Comparator 'x' Event bit  
1 = Generate a Digital Comparator 'x' Event when the DCMPHI<15:0> bits ≤ DATA<31:0> bits  
0 = Do not generate an event
- bit 2 **IEHILO**: High/Low Digital Comparator 'x' Event bit  
1 = Generate a Digital Comparator 'x' Event when the DATA<31:0> bits < DCMPHI<15:0> bits  
0 = Do not generate an event
- bit 1 **IELOHI**: Low/High Digital Comparator 'x' Event bit  
1 = Generate a Digital Comparator 'x' Event when the DCMPL0<15:0> bits ≤ DATA<31:0> bits  
0 = Do not generate an event
- bit 0 **IELOLO**: Low/Low Digital Comparator 'x' Event bit  
1 = Generate a Digital Comparator 'x' Event when the DATA<31:0> bits < DCMPL0<15:0> bits  
0 = Do not generate an event

**Note 1:** This setting is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-27: ADCBASE: ADC BASE REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | ADCBASE<15:8>     |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | ADCBASE<7:0>      |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-0      **Unimplemented:** Read as '0'

bit 15-0      **ADCBASE<15:0>:** ADC ISR Base Address bits

This register, when read, contains the base address of the user's ADC ISR jump table. The interrupt vector address is determined by the IRQVS<2:0> bits of the ADCCON1 register specifying the amount of left shift done to the AIRDYx status bits in the ADCDSTAT1 and ADCDSTAT2 registers, prior to adding with ADCBASE register.

Interrupt Vector Address = Read Value of ADCBASE and Read Value of ADCBASE = Value written to ADCBASE + x << IRQVS<2:0>, where 'x' is the smallest active analog input ID from the ADCDSTAT1 or ADCDSTAT2 registers (which has highest priority).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-28: ADCDSTAT: ADC DMA STATUS REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R/W-0             | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DMAEN             | ---               | RBFIE5            | RBFIE4            | RBFIE3            | RBFIE2            | RBFIE1           | RBFIE0           |
| 23:16     | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC      | R-0, HS, HC      |
|           | WOVERR            | ---               | RBF5              | RBF4              | RBF3              | RBF2              | RBF1             | RBF0             |
| 15:8      | R/W-0             | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DMACEN            | ---               | RAFIE5            | RAFIE4            | RAFIE3            | RAFIE2            | RAFIE1           | RAFIE0           |
| 7:0       | U-0               | U-0               | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC      | R-0, HS, HC      |
|           | ---               | ---               | RAF5              | RAF4              | RAF3              | RAF2              | RAF1             | RAF0             |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31      **DMAEN:** Global ADC DMA Enable bit

1 = DMA interface is enabled  
 0 = DMA interface is disabled

When DMAEN = 0, no data is being saved into internal SRAM, no SRAM Writes occur and the DMA interface logic is being kept in reset state.

**Note:** Before setting the DMAEN bit to '1', the user application must ensure that the BCHEN bit (ADCxTIME<23>) is configured as needed.

bit 30      **Unimplemented:** Read as '0'

bit 29-24      **RBFIE5:RBFIE0:** RAM DMA Buffer B Full Interrupt Enable bits for ADC5-ADC0

1 = Enable ping-pong DMA Buffer B interrupt requests for ADC5-ADC0  
 0 = Disable ping-pong DMA Buffer B interrupt requests for ADC5-ADC0

bit 23      **WOVERR:** DMA Transfer error

This bit is set by hardware and cleared by hardware after a software read of the ADCDSTAT register. If this bit is set, the ADC conversion results transferred by DMA are erroneous. Discarding the entire ADC ram buffer data is recommended.

bit 22      **Unimplemented:** Read as '0'

bit 21-16      **RBF5:RBF0:** RAM DMA Buffer B Full Status bits for ADC5-ADC0

1 = RAM DMA ping-pong Buffer B is full  
 0 = RAM DMA pin-pong Buffer B is not full

When RBFIE<sub>x</sub> = 1 and the RBF<sub>x</sub> bit status is set, the individual ADC<sub>x</sub> DMA interrupt request is generated.

**NOTE:** All of these bits will self-clear upon any read by software of this register. Therefore, It is recommended that the user logically OR the register contents to a user defined variable to be used for testing the DMA status in the user's code. The user software would therefore need to clear the corresponding variable bit after the buffer full status was processed.

bit 15      **DMACEN:** ADC DMA Buffer Sample Count Enable bit

The DMA interface will save the current sample count for each buffer in the table starting at the ADCCNTB address after each sample write into the corresponding buffer in the SRAM.

bit 14      **Unimplemented:** Read as '0'

bit 13-8      **RAFIE5:RAFIE0:** RAM DMA Buffer A Full Interrupt Enable bits for ADC5-ADC0

1 = Enable ping-pong DMA Buffer A interrupt requests for ADC5-ADC0  
 0 = Disable ping-pong DMA Buffer A interrupt requests for ADC5-ADC0

bit 7-6      **Unimplemented:** Read as '0'

**Note:** The individual Class 1 High-Speed ADC5-ADC0 modules have an independent DMA bus master and are completely separate from the assignable general purpose DMA channels.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-28: ADCDSTAT: ADC DMA STATUS REGISTER (CONTINUED)

bit 5-0      **RAF5:RAF0:** RAM DMA Ping-Pong Buffer A Full Status bits for ADC5-ADC0  
1 = RAM DMA ping-pong Buffer A is full  
0 = RAM DMA ping-pong Buffer A is not full

When RAFIEx = 1 and the RAFx bit status is set, the individual ADCx DMA interrupt request is generated.

**NOTE:** All of these bits will self-clear upon any read by software of this register. Therefore, it is recommended that the user logically OR the register contents to a user defined variable to be used for testing the DMA status in the users code. The user software would therefore need to clear the corresponding variable bit after the buffer full status was processed.

**Note:** The individual Class 1 High-Speed ADC5-ADC0 modules have an independent DMA bus master and are completely separate from the assignable general purpose DMA channels.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-29: ADCCNTB: ADC CHANNEL SAMPLE COUNT BASE ADDRESS REGISTER

| Bit Range      | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCCNTB<31:24> |                |                |                |                |                |                |               |               |
| 23:16          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCCNTB<23:16> |                |                |                |                |                |                |               |               |
| 15:8           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCCNTB<15:8>  |                |                |                |                |                |                |               |               |
| 7:0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCCNTB<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-0 **ADCCNTB<31:0>**: ADC Channel Count Base Address bits

SRAM address for the DMA interface at which to save the first class channel buffer A sample count values into the System RAM. If First Class Channel 'x' (where 'x' = 0-5), is ready with a new available sample data, and the DMA interface is currently saving data for Channel 'x' to RAM Buffer 'z' (where 'z' == 0 means Buffer A and 'z' == 1 means Buffer B, with 'z' depending on 'x'), the DMA interface will increment (+1) the 1 byte count value stored at System RAM address (ADCCNTB + 2 \* x + z). ADCCNTB works in conjunction with ADCDMAB. The DMA interface will use ADCCNTB to save the buffer sample counts only if the DMACEN bit in the ADCDSTAT register is set to '1'.

## REGISTER 23-30: ADCDMAB: ADC CHANNEL SAMPLE COUNT BASE ADDRESS REGISTER

| Bit Range      | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCDMAB<31:24> |                |                |                |                |                |                |               |               |
| 23:16          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCDMAB<23:16> |                |                |                |                |                |                |               |               |
| 15:8           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCDMAB<15:8>  |                |                |                |                |                |                |               |               |
| 7:0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| ADCDMAB<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-0 **ADCDMAB<31:0>**: DMA Interface Base Address bits

Address at which to save first class channels data into the System RAM. If First Class Channel 'x' (where 'x' = 0-5), is ready with a new available sample data, and the DMA interface is currently saving data for Channel 'x' to RAM Buffer 'z' (where 'z' == 0 means Buffer A and 'z' == 1 means Buffer B, 'z' depending on 'x'), and the current DMA x-counter value is 'y' (with 'y' depending on 'x'), the DMA interface will store the 2-byte output data value at System RAM address (ADCDMAB + (2 \* x + z) \* 2<sup>(DMABL+1)</sup> + 2 \* y). Also, if the DMACEN bit in the ADCDSTAT register is set to '1', the DMA interface will store without delay the value 'y' itself at the System RAM address (ADCCNTB + 2 \* x + z).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-31: ADCDATAx: ADC OUTPUT DATA REGISTER ('x' = 0-27, 33-41, and 45-53)

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | DATA<31:24>       |                   |                   |                   |                   |                   |                  |                  |
| 23:16     | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | DATA<23:16>       |                   |                   |                   |                   |                   |                  |                  |
| 15:8      | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | DATA<15:8>        |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R-0               | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | DATA<7:0>         |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-0      **DATA<31:0>**: ADC Converted Data Output bits.

- Note 1:** The registers, ADCDATA23-20, ADCDATA41-33, and ADCDATA45-47, are not available on 64-pin devices.
- 2:** The registers, ADCDATA32-28 and ADCDATA44-42, are not available on 64-pin and 100-pin devices.
- 3:** When an alternate input is used as the input source for a dedicated ADC module, the data output is still read from the Primary input Data Output Register.
- 4:** Reading the ADCDATAx register value after changing the FRACT bit converts the data into the format specified by FRACT bit.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-32: ADCTRGNSNS: ADC TRIGGER LEVEL/EDGE SENSITIVITY REGISTER

| Bit Range | Bit<br>31/23/15/7    | Bit<br>30/22/14/6    | Bit<br>29/21/13/5    | Bit<br>28/20/12/4    | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|----------------------|----------------------|----------------------|----------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0                  | U-0                  | U-0                  | U-0                  | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                    | —                    | —                    | —                    | LVL27             | LVL26             | LVL25            | LVL24            |
| 23:16     | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | LVL23 <sup>(1)</sup> | LVL22 <sup>(1)</sup> | LVL21 <sup>(1)</sup> | LVL20 <sup>(1)</sup> | LVL19             | LVL18             | LVL17            | LVL16            |
| 15:8      | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | LVL15                | LVL14                | LVL13                | LVL12                | LVL11             | LVL10             | LVL9             | LVL8             |
| 7:0       | R/W-0                | R/W-0                | R/W-0                | R/W-0                | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | LVL7                 | LVL6                 | LVL5                 | LVL4                 | LVL3              | LVL2              | LVL1             | LVL0             |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-28    **Unimplemented:** Read as '0'

bit 27-0    **LVL27:LVL0:** Trigger Level and Edge Sensitivity bits

- 1 = Analog input is sensitive to the high level of its trigger (level sensitivity implies retriggering as long as the trigger signal remains high)
- 0 = Analog input is sensitive to the positive edge of its trigger (this is the value after a reset)

**Note 1:** This bit is not available on 64-pin devices.

**Note 1:** This register specifies the trigger level for analog inputs 0 to 27.

**2:** The higher analog input ID belongs to Class 3, and therefore, is only scan triggered. All Class 3 analog inputs use the Scan Trigger, for which the level/edge is defined by the STRGLVL bit (ADCCON1<3>).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-33: ADCxTIME: DEDICATED HIGH-SPEED ADCx TIMING REGISTER (‘x’ = 0 THROUGH 5)

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-1            | R/W-1            |
|           | —                 | —                 | —                 | ADCEIS<2:0>       |                   |                   | SELRES<1:0>      |                  |
| 23:16     | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | BCHEN             | ADCDIV<6:0>       |                   |                   |                   |                   |                  |                  |
| 15:8      | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | R/W-0            | R/W-0            |
|           | —                 | —                 | —                 | —                 | —                 | —                 | SAMC<9:8>        |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | SAMC<7:0>         |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as ‘0’  
 -n = Value at POR      ‘1’ = Bit is set      ‘0’ = Bit is cleared      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as ‘0’

bit 28-26 **ADCEIS<2:0>**: ADCx Early Interrupt Select bits

111 = The data ready interrupt is generated 8 ADC clocks prior to the end of conversion

110 = The data ready interrupt is generated 7 ADC clocks prior to the end of conversion

⋮

⋮

001 = The data ready interrupt is generated 2 ADC clocks prior to the end of conversion

000 = The data ready interrupt is generated 1 ADC clock prior to the end of conversion

**Note:** All options are available when the selected resolution, specified by the SELRES<1:0> bits (ADCxTIME<25:24>), is 12-bit or 10-bit. For a selected resolution of 8-bit, options from ‘000’ to ‘101’ are valid. For a selected resolution of 6-bit, options from ‘000’ to ‘011’ are valid.

bit 25-24 **SELRES<1:0>**: ADCx Resolution Select bits

11 = 12 bits

10 = 10 bits

01 = 8 bits

00 = 6 bits

bit 23 **BCHEN**: Buffer Channel Enable bit

1 = ADC data saved in DMA system ram buffer when DMAEN (ADCDSTAT<31>) = 1

0 = ADC data must be read by CPU from appropriate ADC result register

bit 22-16 **ADCDIV<6:0>**: ADCx Clock Divisor bits

These bits divide the ADC control clock with period T<sub>Q</sub> to generate the clock for ADCx (T<sub>ADx</sub>).

1111111 = 254 \* T<sub>Q</sub> = T<sub>ADx</sub>

⋮

⋮

0000011 = 6 \* T<sub>Q</sub> = T<sub>ADx</sub>

0000010 = 4 \* T<sub>Q</sub> = T<sub>ADx</sub>

0000001 = 2 \* T<sub>Q</sub> = T<sub>ADx</sub>

0000000 = Reserved

bit 15-10 **Unimplemented:** Read as ‘0’



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-33: ADCxTIME: DEDICATED HIGH-SPEED ADCx TIMING REGISTER (CONTINUED) (‘x’ = 0 THROUGH 5)

bit 9-0 **SAMC<9:0>**: ADCx Sample Time bits

Where TADx = period of the ADC conversion clock for the dedicated ADC controlled by the ADCDIV<6:0> bits.

1111111111 = 1025 TADx

•

•

0000000001 = 3 TADx

0000000000 = 2 TADx

**Note:** The SAMC sample time is always enforced regardless even if the conversion trigger occurs before SAMC expiration. The conversion trigger event is persistent and will be acknowledged and start the conversion if true, immediately after the SAMC period. ADC0-ADC5 will remain indefinitely in the sample state even after the expiration of SAMC until the trigger event, which will end sampling and start conversion, except when either of the following are true:

-The ADC filter is enabled and the DFMODE bit in the ADCFLTRx register = 0

-The TRGSRC3 bit in the ADCTRG1 register = Global level software trigger

**Note:** If ADCFLTRx<AFEN> = 1, then the ADCxTIME<SAMC> min must be  $\geq$  to 0x004 or equivalent to  $\geq$  6 TADx. This will correspondingly reduce the max sampling rate possible.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-34: ADCEIEN1: ADC EARLY INTERRUPT ENABLE REGISTER 1

| Bit Range | Bit 31/23/15/7        | Bit 30/22/14/6        | Bit 29/21/13/5        | Bit 28/20/12/4        | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                   | U-0                   | U-0                   | U-0                   | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —                     | —                     | —                     | —                     | EIEN27         | EIEN26         | EIEN25        | EIEN24        |
| 23:16     | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EIEN23 <sup>(1)</sup> | EIEN22 <sup>(1)</sup> | EIEN21 <sup>(1)</sup> | EIEN20 <sup>(1)</sup> | EIEN19         | EIEN18         | EIEN17        | EIEN16        |
| 15:8      | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EIEN15                | EIEN14                | EIEN13                | EIEN12                | EIEN11         | EIEN10         | EIEN9         | EIEN8         |
| 7:0       | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EIEN7                 | EIEN6                 | EIEN5                 | EIEN4                 | EIEN3          | EIEN2          | EIEN1         | EIEN0         |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | C = Clearable bit                            |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **EIEN27:EIEN0:** Early Interrupt Enable for Analog Input bits

- 1 = Early Interrupts are enabled for the selected analog input. The interrupt is generated after the early interrupt event occurs (indicated by the EIRDYx bit ('x' = 31-0) of the ADCEIEN1 register)
- 0 = Interrupts are disabled

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-35: ADCEIEN2: ADC EARLY INTERRUPT ENABLE REGISTER 2

| Bit Range | Bit 31/23/15/7                 | Bit 30/22/14/6                 | Bit 29/21/13/5                  | Bit 28/20/12/4                 | Bit 27/19/11/3                 | Bit 26/18/10/2                 | Bit 25/17/9/1                  | Bit 24/16/8/0                  |
|-----------|--------------------------------|--------------------------------|---------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| 31:24     | U-0<br>—                       | U-0<br>—                       | U-0<br>—                        | U-0<br>—                       | U-0<br>—                       | U-0<br>—                       | U-0<br>—                       | U-0<br>—                       |
| 23:16     | U-0<br>—                       | U-0<br>—                       | R/W-0<br>EIEN53                 | R/W-0<br>EIEN52                | R/W-0<br>EIEN51                | R/W-0<br>EIEN50                | R/W-0<br>EIEN49                | R/W-0<br>EIEN48                |
| 15:8      | R/W-0<br>EIEN47 <sup>(1)</sup> | R/W-0<br>EIEN46 <sup>(1)</sup> | R/W-0<br>EIEN45 <sup>(1)</sup>  | U-0<br>—                       | U-0<br>—                       | U-0<br>—                       | R/W-0<br>EIEN41 <sup>(1)</sup> | R/W-0<br>EIEN40 <sup>(1)</sup> |
| 7:0       | R/W-0<br>EIEN39 <sup>(1)</sup> | R/W-0<br>EIEN38 <sup>(1)</sup> | R/W-0<br>EIEN37 <sup>(12)</sup> | R/W-0<br>EIEN36 <sup>(1)</sup> | R/W-0<br>EIEN35 <sup>(1)</sup> | R/W-0<br>EIEN34 <sup>(1)</sup> | R/W-0<br>EIEN33 <sup>(1)</sup> | U-0<br>—                       |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | C = Clearable bit                            |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-22 **Unimplemented:** Read as '0'

bit 21-13 **EIEN53:EIEN45:** Early Interrupt Enable for Analog Input bits

- 1 = Early Interrupts are enabled for the selected analog input. The interrupt is generated after the early interrupt event occurs (indicated by the EIRDYx bit ('x' = 44-32) of the ADCEIEN2 register)
- 0 = Interrupts are disabled

bit 12-10 **Unimplemented:** Read as '0'

bit 9-1 **EIEN41:EIEN33:** Early Interrupt Enable for Analog Input bits

- 1 = Early Interrupts are enabled for the selected analog input. The interrupt is generated after the early interrupt event occurs (indicated by the EIRDYx bit ('x' = 44-32) of the ADCEIEN2 register)
- 0 = Interrupts are disabled

bit 0 **Unimplemented:** Read as '0'

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-36: ADCEI1STAT1: ADC EARLY INTERRUPT STATUS REGISTER 1

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6         | Bit 29/21/13/5         | Bit 28/20/12/4         | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|------------------------|------------------------|------------------------|------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                    | U-0                    | U-0                    | U-0                    | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | —                      | —                      | —                      | —                      | EIRDY27        | EIRDY26        | EIRDY25       | EIRDY24       |
| 23:16     | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | EIRDY23 <sup>(1)</sup> | EIRDY22 <sup>(1)</sup> | EIRDY21 <sup>(1)</sup> | EIRDY20 <sup>(1)</sup> | EIRDY19        | EIRDY18        | EIRDY17       | EIRDY16       |
| 15:8      | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | EIRDY15                | EIRDY14                | EIRDY13                | EIRDY12                | EIRDY11        | EIRDY10        | EIRDY9        | EIRDY8        |
| 7:0       | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC            | R-0, HS, HC    | R-0, HS, HC    | R-0, HS, HC   | R-0, HS, HC   |
|           | EIRDY7                 | EIRDY6                 | EIRDY5                 | EIRDY4                 | EIRDY3         | EIRDY2         | EIRDY1        | EIRDY0        |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Cleared by hardware                     |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **EIRDY27:EIRDY0:** Early Interrupt for Corresponding Analog Input Ready bits

- 1 = This bit is set when the early interrupt event occurs for the specified analog input. An interrupt will be generated if early interrupts are enabled in the ADCEIEN1 register. For the Class 1 analog inputs, this bit will set as per the configuration of the ADCEIS<2:0> bits in the ADCxTIME register. For the shared ADC module, this bit will be set as per the configuration of the ADCEIS<2:0> bits in the ADCCON2 register.
- 0 = Interrupts are disabled

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-37: ADCEIEN2: ADC EARLY INTERRUPT STATUS REGISTER 2

| Bit Range | Bit 31/23/15/7                        | Bit 30/22/14/6                        | Bit 29/21/13/5                        | Bit 28/20/12/4                        | Bit 27/19/11/3                        | Bit 26/18/10/2                        | Bit 25/17/9/1                         | Bit 24/16/8/0                         |
|-----------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|
| 31:24     | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              |
| 23:16     | U-0<br>—                              | U-0<br>—                              | R-0, HS, HC<br>EIRDY53                | R-0, HS, HC<br>EIRDY52                | R-0, HS, HC<br>EIRDY51                | R-0, HS, HC<br>EIRDY50                | R-0, HS, HC<br>EIRDY49                | R-0, HS, HC<br>EIRDY48                |
| 15:8      | R-0, HS, HC<br>EIRDY47 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY46 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY45 <sup>(1)</sup> | U-0<br>—                              | U-0<br>—                              | U-0<br>—                              | R-0, HS, HC<br>EIRDY41 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY40 <sup>(1)</sup> |
| 7:0       | R-0, HS, HC<br>EIRDY39 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY38 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY37 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY36 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY35 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY34 <sup>(1)</sup> | R-0, HS, HC<br>EIRDY33 <sup>(1)</sup> | U-0<br>—                              |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Cleared by hardware                     |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-22 **Unimplemented:** Read as '0'

bit 21-13 **EIRDY53:EIRDY45:** Early Interrupt for Corresponding Analog Input Ready bits

- 1 = This bit is set when the early interrupt event occurs for the specified analog input. An interrupt will be generated if early interrupts are enabled in the ADCEIEN2 register. For the Class 1 analog inputs, this bit will set as per the configuration of the ADCEIS<2:0> bits in the ADCxTIME register. For the shared ADC module, this bit will be set as per the configuration of the ADCEIS<2:0> bits in the ADCCON2 register.
- 0 = Interrupts are disabled

bit 12-10 **Unimplemented:** Read as '0'

bit 9-1 **EIRDY41:EIRDY33:** Early Interrupt for Corresponding Analog Input Ready bits

- 1 = This bit is set when the early interrupt event occurs for the specified analog input. An interrupt will be generated if early interrupts are enabled in the ADCEIEN2 register. For the Class 1 analog inputs, this bit will set as per the configuration of the ADCEIS<2:0> bits in the ADCxTIME register. For the shared ADC module, this bit will be set as per the configuration of the ADCEIS<2:0> bits in the ADCCON2 register.
- 0 = Interrupts are disabled

bit 0 **Unimplemented:** Read as '0'

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-38: ADCANCON: ADC ANALOG WARM-UP CONTROL REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | —                 | —                 | WKUPCLKCNT<3:0>   |                   |                  |                  |
| 23:16     | R/W-0             | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | WKIEN7            | —                 | WKIEN5            | WKIEN4            | WKIEN3            | WKIEN2            | WKIEN1           | WKIEN0           |
| 15:8      | R-0, HS, HC       | U-0               | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC       | R-0, HS, HC      | R-0, HS, HC      |
|           | WKRDY7            | —                 | WKRDY5            | WKRDY4            | WKRDY3            | WKRDY2            | WKRDY1           | WKRDY0           |
| 7:0       | R/W-0             | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | ANEN7             | —                 | ANEN5             | ANEN4             | ANEN3             | ANEN2             | ANEN1            | ANEN0            |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Cleared by Software                     |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-24 **WKUPCLKCNT<3:0>:** Wake-up Clock Count bits

These bits represent the number of ADC clocks required to warm-up the ADC module before it can perform conversion. Although the clocks are specific to each ADC, the WKUPCLKCNT bit is common to all ADC modules.

1111 =  $2^{15}$  = 32,768 clocks

•  
•

0110 =  $2^6$  = 64 clocks

0101 =  $2^5$  = 32 clocks

0100 =  $2^4$  = 16 clocks

0011 =  $2^4$  = 16 clocks

0010 =  $2^4$  = 16 clocks

0001 =  $2^4$  = 16 clocks

0000 =  $2^4$  = 16 clocks

**Note:** Minimum required ADCx warm-up time, (i.e., WKUPCLKCNT), is the lesser of 500 ADC clocks, (i.e., TAD), or 20  $\mu$ s.

bit 23 **WKIEN7:** Shared ADC (ADC7) Wake-up Interrupt Enable bit

1 = Enable interrupt and generate interrupt when the WKRDY7 status bit is set

0 = Disable interrupt

bit 22 **Unimplemented:** Read as '0'

bit 21-16 **WKIEN5:WKIEN0:** ADC5-ADC0 Wake-up Interrupt Enable bit

1 = Enable interrupt and generate interrupt when the WKRDYx status bit is set

0 = Disable interrupt

bit 15 **WKRDY7:** Shared ADC (ADC7) Wake-up Status bit

1 = ADC7 Analog and Bias circuitry ready after the wake-up count number  $2^{WKUPEXP}$  clocks after setting ANEN7 to '1'

0 = ADC7 Analog and Bias circuitry is not ready

**Note:** This bit is cleared by hardware when the ANEN7 bit is cleared

bit 14 **Unimplemented:** Read as '0'

bit 13-8 **WKRDY5:WKRDY0:** ADC5-ADC0 Wake-up Status bit

1 = ADCx Analog and Bias circuitry ready after the wake-up count number  $2^{WKUPEXP}$  clocks after setting ANENx to '1'

0 = ADCx Analog and Bias circuitry is not ready

**Note:** These bits are cleared by hardware when the ANENx bit is cleared

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 23-38: ADCANCON: ADC ANALOG WARM-UP CONTROL REGISTER (CONTINUED)

- bit 7      **ANEN7:** Shared ADC (ADC7) Analog and Bias Circuitry Enable bit  
1 = Analog and bias circuitry enabled. Once the analog and bias circuit is enabled, the ADC module needs a warm-up time, as defined by the WKUPCLKCNT<3:0> bits.  
0 = Analog and bias circuitry disabled
- bit 6      **Unimplemented:** Read as '0'
- bit 5-0    **ANEN5:ANEN0:** ADC5-ADC0 Analog and Bias Circuitry Enable bits  
1 = Analog and bias circuitry enabled. Once the analog and bias circuit is enabled, the ADC module needs a warm-up time, as defined by the WKUPCLKCNT<3:0> bits.  
0 = Analog and bias circuitry disabled

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-39: ADCxCFG: ADCx CONFIGURATION REGISTER ('x' = 0 THROUGH 5 and 7)

| Bit Range     | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|---------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24         | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
| ADCCFG<31:24> |                   |                   |                   |                   |                   |                   |                  |                  |
| 23:16         | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
| ADCCFG<23:16> |                   |                   |                   |                   |                   |                   |                  |                  |
| 15:8          | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
| ADCCFG<15:8>  |                   |                   |                   |                   |                   |                   |                  |                  |
| 7:0           | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
| ADCCFG<7:0>   |                   |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 31-0      **ADCCFG<31:0>**: ADC Module Configuration Data bits

**Note:** These bits can only change when the applicable ANENx bit in the ADCANCON register is cleared. These are calibration values determined at product test time and are provided to the user to copy and write into these registers.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-40: ADCSYSCFG0: ADC SYSTEM CONFIGURATION REGISTER 0

| Bit Range | Bit<br>31/23/15/7   | Bit<br>30/22/14/6   | Bit<br>29/21/13/5   | Bit<br>28/20/12/4   | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|---------------------|---------------------|---------------------|---------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0                 | U-0                 | U-0                 | U-0                 | HC, HS, R-0       | HC, HS, R-0       | HC, HS, R-0      | HC, HS, R-0      |
|           | —                   | —                   | —                   | —                   | AN27              | AN26              | AN25             | AN24             |
| 23:16     | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0       | HC, HS, R-0       | HC, HS, R-0      | HC, HS, R-0      |
|           | AN23 <sup>(1)</sup> | AN22 <sup>(1)</sup> | AN21 <sup>(1)</sup> | AN20 <sup>(1)</sup> | AN19              | AN18              | AN17             | AN16             |
| 15:8      | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0       | HC, HS, R-0       | HC, HS, R-0      | HC, HS, R-0      |
|           | AN15                | AN14                | AN13                | AN12                | AN11              | AN10              | AN9              | AN8              |
| 7:0       | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0       | HC, HS, R-0       | HC, HS, R-0      | HC, HS, R-0      |
|           | AN7                 | AN6                 | AN5                 | AN4                 | AN3               | AN2               | AN1              | AN0              |

|                   |                   |  |
|-------------------|-------------------|--|
| <b>Legend:</b>    | HS = Hardware Set | HC = Cleared by Software                     |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-0 **AN27:AN0>:** ADC Analog Input bits

These bits reflect the system configuration and are updated during boot-up time. By reading these read-only bits, the user application can determine whether or not an analog input in the device is available.

AN<31:0>: Reflects the presence or absence of the respective analog input (AN31-AN0).

**Note 1:** This bit is not available on 64-pin devices.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 23-41: ADCSYSCFG1: ADC SYSTEM CONFIGURATION REGISTER 1

| Bit Range | Bit<br>31/23/15/7   | Bit<br>30/22/14/6   | Bit<br>29/21/13/5   | Bit<br>28/20/12/4   | Bit<br>27/19/11/3   | Bit<br>26/18/10/2   | Bit<br>25/17/9/1    | Bit<br>24/16/8/0    |
|-----------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| 31:24     | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 | U-0                 |
|           | —                   | —                   | —                   | —                   | —                   | —                   | —                   | —                   |
| 23:16     | U-0                 | U-0                 | HC, HS, R-0         | r-0                 | r-1                 | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         |
|           | —                   | —                   | AN53 <sup>(2)</sup> | —                   | —                   | AN50 <sup>(2)</sup> | AN49                | AN48                |
| 15:8      | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | U-0                 | U-0                 | U-0                 | HC, HS, R-0         | HC, HS, R-0         |
|           | AN47 <sup>(1)</sup> | AN46 <sup>(1)</sup> | AN45 <sup>(1)</sup> | —                   | —                   | —                   | AN41 <sup>(1)</sup> | AN40 <sup>(1)</sup> |
| 7:0       | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | HC, HS, R-0         | U-0                 |
|           | AN39 <sup>(1)</sup> | AN38 <sup>(1)</sup> | AN37 <sup>(1)</sup> | AN36 <sup>(1)</sup> | AN35 <sup>(1)</sup> | AN34 <sup>(1)</sup> | AN33 <sup>(1)</sup> | —                   |

|                   |                   |   |
|-------------------|-------------------|---|
| <b>Legend:</b>    | HS = Hardware Set | HC = Cleared by Software                                |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0'      r =Reserved bit |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared      x = Bit is unknown            |

bit 31-22 **Unimplemented:** Read as '0'

bit 21 **AN53:** ADC Analog Input bits

These bits reflect the system configuration and are updated during boot-up time. By reading these read-only bits, the user application can determine whether or not an analog input in the device is available.

AN<63:32>: Reflects the presence or absence of the respective analog input (AN63-AN32).

bit 20-19 **Reserved**

bit 18-13 **AN50:AN45:** ADC Analog Input bits

These bits reflect the system configuration and are updated during boot-up time. By reading these read-only bits, the user application can determine whether or not an analog input in the device is available.

AN<63:32>: Reflects the presence or absence of the respective analog input (AN63-AN32).

bit 12-10 **Unimplemented:** Read as '0'

bit 9-1 **AN41:AN33:** ADC Analog Input bits

These bits reflect the system configuration and are updated during boot-up time. By reading these read-only bits, the user application can determine whether or not an analog input in the device is available.

AN<63:32>: Reflects the presence or absence of the respective analog input (AN63-AN32).

bit 0 **Unimplemented:** Read as '0'

**Note 1:** This bit is not available on 64-pin devices.

**2:** Internal Analog inputs: AN50 = IVREF (1.2V), AN53 = CTMU\_TEMP.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 24.0 CONTROLLER AREA NETWORK WITH FLEXIBLE DATA-RATE (CAN FD)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 56. “Controller Area Network with Flexible Data-rate (CAN FD)”** (*DS number pending*), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Controller Area Network with Flexible Data-rate (CAN FD) module supports the following key features:

- Standards Compliance:
  - Full CAN 2.0B compliance
  - Programmable bit rate
  - ISO11898-1:2015 plus CAN FD 1.0 compliant, supports up to 64 data bytes payload/message
  - Arbitration Bit Rate up to one Mbps
  - FD Bit Rate up to eight Mbps
- Message Reception and Transmission:
  - 32 message FIFOs
  - Each FIFO can have up to 32 messages for a total of 512 messages
  - FIFO can be a transmit message FIFO or a receive message FIFO
  - User-defined priority levels for message FIFOs used for transmission
  - 32 acceptance filters for message filtering
  - 32 acceptance filter mask registers for message filtering
  - Automatic response to remote transmit request
  - DeviceNet™ addressing support

- Additional Features:
  - Loopback, Listen All Messages, and Listen Only modes for self-test, system diagnostics and bus monitoring
  - Low-power operating modes
  - CAN FD module is a dedicated DMA bus master on the PIC32MK system bus so use of system general purpose DMA is not required
  - Dedicated time-stamp timer
  - Data-only Message Reception mode
  - Selective wake-up, and transceiver standby control
  - Low-power operating mode

### CAUTION

The CAN FD protocol was defined to allow CAN 2.0 messages and CAN FD messages to co-exist on the same bus. This does not imply that non-CAN FD controllers can be mixed with CAN FD controllers on the same bus. Non-CAN FD controllers will generate error frames while receiving a CAN FD message

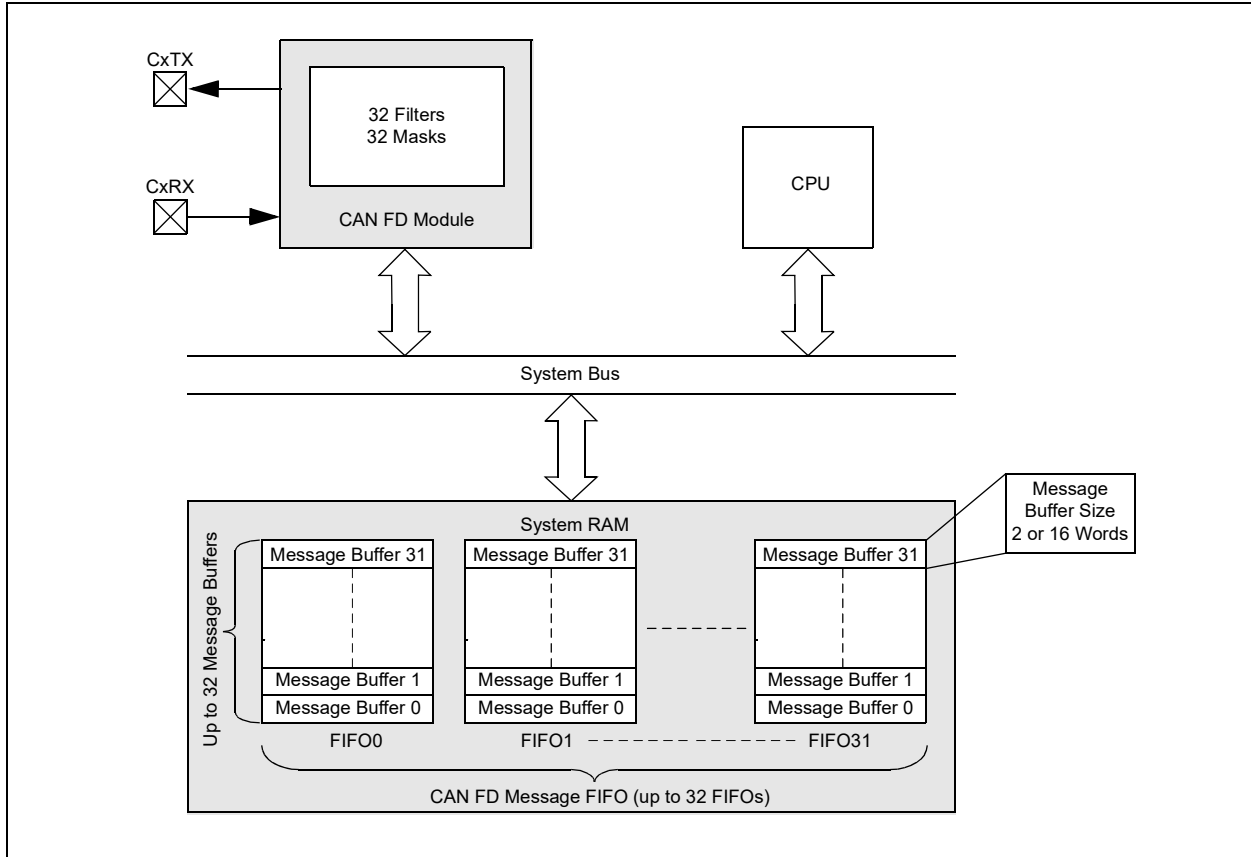
Figure 24-1 illustrates the general structure of the CAN FD module.

**Note:** Recommended CAN-FD clock frequency is 20MHz/40MHz/80MHz. Implementing following steps to set Module Clock frequency of 40MHz while maintaining CPU clock frequency of 120MHz:

- Select module clock source as REF-CLK4 by configuring CLKSEL0 bit (CFDxCON<7>) = 1
- Select SYSCLK as the input clock from REFCLK4 by configure ROSEL bits (REFO4CON<3:0>) = 0
- Set REF4CLK prescaler to 3 by configuring RODIV bits (REFO4CON<30:16>) = 1 and ROTRIM bits (REFO4TRIM<31:23> = 256

# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 24-1: PIC32MK CAN FD MODULE BLOCK DIAGRAM



## 24.1 Control Registers

**TABLE 24-1: CAN FD PERIPHERAL REGISTER SUMMARY**

| Virtual Address (BF88_#) | Register Name | Bit Range | Bits           |             |         |             |            |            |               |             |             |            |            |          |               |         |             | All Resets |       |      |      |
|--------------------------|---------------|-----------|----------------|-------------|---------|-------------|------------|------------|---------------|-------------|-------------|------------|------------|----------|---------------|---------|-------------|------------|-------|------|------|
|                          |               |           | 31/15          | 30/14       | 29/13   | 28/12       | 27/11      | 26/10      | 25/9          | 24/8        | 23/7        | 22/6       | 21/5       | 20/4     | 19/3          | 18/2    | 17/1        |            | 16/0  |      |      |
| 0000                     | CFD1CON       | 31:16     | TXBWS<3:0>     |             |         | ABAT        | REQOP<2:0> |            |               | OPMOD<2:0>  |             |            | TXQEN      | STEF     | SERR2LOM      | ESIGM   | RTXAT       | 0498       |       |      |      |
|                          |               | 15:0      | ON             | —           | SIDL    | BRSDIS      | BUSY       | WFT<1:0>   | WAKFIL        | CLKSEL0     | PXEDIS      | ISOCRCEN   | DNCNT<4:0> |          |               |         |             | 0760       |       |      |      |
| 0010                     | CFD1NBTCFG    | 31:16     | BRP<7:0>       |             |         |             |            |            | TSEG1<7:0>    |             |             | TSEG1<7:0> |            |          |               |         |             | 003E       |       |      |      |
|                          |               | 15:0      | —              | TSEG2<6:0>  |         |             |            |            |               | —           | SJW<6:0>    |            |            |          |               |         | 0F0F        |            |       |      |      |
| 0020                     | CFD1DBTCFG    | 31:16     | BRP<7:0>       |             |         |             |            |            | —             | —           | —           | TSEG1<4:0> |            |          |               |         |             | 000E       |       |      |      |
|                          |               | 15:0      | —              | —           | —       | —           | —          | TSEG2<3:0> |               |             | —           | —          | —          | —        | SJW<3:0>      |         |             | 0303       |       |      |      |
| 0030                     | CFD1TDC       | 31:16     | —              | —           | —       | —           | —          | —          | EDGFLTEN      | SID11EN     | —           | —          | —          | —        | —             | —       | TDCMOD<1:0> | 0002       |       |      |      |
|                          |               | 15:0      | —              | TDCO<6:0>   |         |             |            |            |               | —           | —           | TDCV<5:0>  |            |          |               |         |             | 1000       |       |      |      |
| 0040                     | CFD1TBC       | 31:16     | TBC<31:16>     |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | TBC<15:0>      |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
| 0050                     | CFD1TSCON     | 31:16     | —              | —           | —       | —           | —          | —          | —             | —           | —           | —          | —          | —        | —             | —       | TSRES       | TSEOF      | TBCEN | 0000 |      |
|                          |               | 15:0      | —              | —           | —       | —           | —          | —          | —             | TBCPRE<9:0> |             |            |            |          |               |         |             |            | 0000  |      |      |
| 0060                     | CFD1VEC       | 31:16     | —              | RXCODE<6:0> |         |             |            |            |               | —           | TXCODE<6:0> |            |            |          |               |         | 4040        |            |       |      |      |
|                          |               | 15:0      | —              | —           | —       | FILHIT<4:0> |            |            | —             | ICODE<6:0>  |             |            |            |          |               | 0040    |             |            |       |      |      |
| 0070                     | CFD1INT       | 31:16     | IVMIE          | WAKIE       | CERRIE  | SERRIE      | RXOVIE     | TXATIE     | —             | —           | —           | —          | —          | TEFIE    | MODIE         | TBCIE   | RXIE        | TXIE       | 0000  |      |      |
|                          |               | 15:0      | IVMIF          | WAKIF       | CERRIF  | SERRIF      | RXOVIF     | TXATIF     | —             | —           | —           | —          | —          | TEFIF    | MODIF         | TBCIF   | RXIF        | TXIF       | 0000  |      |      |
| 0080                     | CFD1RXIF      | 31:16     | RFIF<31:16>    |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | RFIF<15:1>     |             |         |             |            |            |               |             |             |            |            |          |               |         |             | —          |       |      |      |
| 0090                     | CFD1TXIF      | 31:16     | TFIF<31:16>    |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | TFIF<15:0>     |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
| 00A0                     | CFD1RXOVIF    | 31:16     | RFOVIF<31:16>  |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | RFOVIF<15:1>   |             |         |             |            |            |               |             |             |            |            |          |               |         |             | —          |       |      |      |
| 00B0                     | CFD1TXATIF    | 31:16     | TFATIF<31:16>  |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | TFATIF<15:0>   |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
| 00C0                     | CFD1TXREQ     | 31:16     | TXREQ<31:16>   |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | TXREQ<15:0>    |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
| 00D0                     | CFD1TREC      | 31:16     | —              | —           | —       | —           | —          | —          | —             | —           | —           | —          | TXBO       | TXBP     | RXBP          | TXWARN  | RXWARN      | EWARN      | 0020  |      |      |
|                          |               | 15:0      | TERRCNT<7:0>   |             |         |             |            |            | RERRCNT<7:0>  |             |             |            |            |          | RERRCNT<7:0>  |         |             |            |       |      | 0000 |
| 00E0                     | CFD1BDIAG0    | 31:16     | DTERRCNT<7:0>  |             |         |             |            |            | DRERRCNT<7:0> |             |             |            |            |          | DRERRCNT<7:0> |         |             |            |       |      | 0000 |
|                          |               | 15:0      | NTERRCNT<7:0>  |             |         |             |            |            | NRERRCNT<7:0> |             |             |            |            |          | NRERRCNT<7:0> |         |             |            |       |      | 0000 |
| 00F0                     | CFD1BDIAG1    | 31:16     | DLCMM          | ESI         | DCRCERR | DSTUFERR    | DFORMERR   | —          | DBIT1ERR      | DBIT0ERR    | —           | —          | NCRCERR    | NSTUFERR | NFORMERR      | NACKERR | NBIT1ERR    | NBIT0ERR   | 0000  |      |      |
|                          |               | 15:0      | EFMSGCNT<15:0> |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
| 0100                     | CFD1TEFCON    | 31:16     | —              | —           | —       | FSIZE<4:0>  |            |            | —             | —           | —           | —          | —          | —        | —             | —       | —           | —          | 0000  |      |      |
|                          |               | 15:0      | —              | —           | —       | —           | —          | FRESET     | —             | UINC        | —           | —          | TEFTSEN    | —        | TEFOVIE       | TEFFIE  | TEFHIE      | TEFNEIE    | 0400  |      |      |
| 0110                     | CFD1TEFSTA    | 31:16     | —              | —           | —       | —           | —          | —          | —             | —           | —           | —          | —          | —        | —             | —       | —           | —          | 0000  |      |      |
|                          |               | 15:0      | —              | —           | —       | —           | —          | —          | —             | —           | —           | —          | —          | —        | TEFOVIF       | TEFFIF  | TEFHIF      | TEFNEIF    | 0000  |      |      |
| 0120                     | CFD1TEFUA     | 31:16     | TEFUA<31:16>   |             |         |             |            |            |               |             |             |            |            |          |               |         |             | xxxx       |       |      |      |
|                          |               | 15:0      | TEFUA<15:0>    |             |         |             |            |            |               |             |             |            |            |          |               |         |             | xxxx       |       |      |      |
| 0130                     | CFD1FIFOPA    | 31:16     | FIFOPA<31:16>  |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |
|                          |               | 15:0      | FIFOPA<15:0>   |             |         |             |            |            |               |             |             |            |            |          |               |         |             | 0000       |       |      |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note 1:** The lower order byte of the 32-bit register resides at the low-order address.

**TABLE 24-1: CAN FD PERIPHERAL REGISTER SUMMARY (CONTINUED)**

| Virtual Address<br>(BF88_#)   | Register Name                  | Bit Range     | Bits          |       |        |       |            |       |        |       |            |            |       |            |        |          |          |          | All Resets |            |        |      |
|-------------------------------|--------------------------------|---------------|---------------|-------|--------|-------|------------|-------|--------|-------|------------|------------|-------|------------|--------|----------|----------|----------|------------|------------|--------|------|
|                               |                                |               | 31/15         | 30/14 | 29/13  | 28/12 | 27/11      | 26/10 | 25/9   | 24/8  | 23/7       | 22/6       | 21/5  | 20/4       | 19/3   | 18/2     | 17/1     | 16/0     |            |            |        |      |
| 0140                          | CFD1TXQCON                     | 31:16         | PLSIZE<2:0>   |       |        |       | FSIZE<4:0> |       |        |       | TXAT<1:0>  |            |       | TXPRI<4:0> |        |          |          |          | 0060       |            |        |      |
|                               |                                | 15:0          | —             | —     | —      | —     | —          | —     | FRESET | TXREQ | UINC       | TXEN       | —     | —          | TXATIE | —        | —        | TXQEIE   | —          | —          | TXQNie | 0480 |
| 0150                          | CFD1TXQSTA                     | 31:16         | —             | —     | —      | —     | —          | —     | —      | —     | —          | —          | —     | —          | —      | —        | —        | —        | —          | —          | —      | 0000 |
|                               |                                | 15:0          | TXQCI<4:0>    |       |        |       |            |       |        |       | TXABT      | TXLARB     | TXERR | TXATIF     | —      | TXQEIF   | —        | —        | —          | —          | TXQNI  | 0000 |
| 0160                          | CFD1TXQUA                      | 31:16         | TXQUA<31:16>  |       |        |       |            |       |        |       |            |            |       |            |        |          |          |          | xxxx       |            |        |      |
|                               |                                | 15:0          | TXQUA<15:0>   |       |        |       |            |       |        |       |            |            |       |            |        |          |          |          | xxxx       |            |        |      |
| 0170                          | CFD1FIFOCONn<br>(n' = 1)       | 31:16         | PLSIZE<2:0>   |       |        |       | FSIZE<4:0> |       |        |       | TXAT<1:0>  |            |       | TXPRI<4:0> |        |          |          |          | 0060       |            |        |      |
|                               |                                | 15:0          | —             | —     | —      | —     | —          | —     | FRESET | TXREQ | UINC       | TXEN       | RTREN | RXTSEN     | TXATIE | RXOVIE   | TFERFFIE | TFHRFHIE | TFNRFNIE   | —          | —      | 0400 |
| 0180                          | CFD1FIFOStAn<br>(n' = 1)       | 31:16         | —             | —     | —      | —     | —          | —     | —      | —     | —          | —          | —     | —          | —      | —        | —        | —        | —          | —          | —      | 0000 |
|                               |                                | 15:0          | FIFOCl<4:0>   |       |        |       |            |       |        |       | TXABT      | TXLARB     | TXERR | TXATIF     | RXOVIF | TFERFFIF | TFHRFHIF | TFNRFNIF | —          | —          | 0000   |      |
| 0190                          | CFD1FIFOuAn<br>(n' = 1)        | 31:16         | FIFOuA<31:16> |       |        |       |            |       |        |       |            |            |       |            |        |          |          |          | xxxx       |            |        |      |
|                               |                                | 15:0          | FIFOuA<15:0>  |       |        |       |            |       |        |       |            |            |       |            |        |          |          |          | xxxx       |            |        |      |
| 01A0-<br>0730                 | CFD1FIFOCONn<br>(n' = 2-31)    | 31:16         | PLSIZE<2:0>   |       |        |       | FSIZE<4:0> |       |        |       | TXAT<1:0>  |            |       | TXPRI<4:0> |        |          |          |          | 0000       |            |        |      |
|                               |                                | 15:0          | —             | —     | —      | —     | —          | —     | FRESET | TXREQ | UINC       | TXEN       | RTREN | RXTSEN     | TXATIE | RXOVIE   | TFERFFIE | TFHRFHIE | TFNRFNIE   | —          | —      | 0000 |
|                               | CFD1FIFOStAn<br>(n' = 2 to 31) | 31:16         | —             | —     | —      | —     | —          | —     | —      | —     | —          | —          | —     | —          | —      | —        | —        | —        | —          | —          | —      | 0000 |
|                               |                                | 15:0          | FIFOCl<4:0>   |       |        |       |            |       |        |       | TXABT      | TXLARB     | TXERR | TXATIF     | RXOVIF | TFERFFIF | TFHRFHIF | TFNRFNIF | —          | —          | 0000   |      |
| CFD1FIFOuAn<br>(n' = 2 to 31) | 31:16                          | FIFOuA<31:16> |               |       |        |       |            |       |        |       |            |            |       |            |        |          |          | 0000     |            |            |        |      |
|                               | 15:0                           | FIFOuA<15:0>  |               |       |        |       |            |       |        |       |            |            |       |            |        |          |          | 0000     |            |            |        |      |
| 0740                          | CFD1FLtCON0                    | 31:16         | FLTEN3        | —     | —      | —     | —          | —     | —      | —     | F3BP<4:0>  | FLTEN2     | —     | —          | —      | —        | —        | —        | —          | F2BP<4:0>  | 0000   |      |
|                               |                                | 15:0          | FLTEN1        | —     | —      | —     | —          | —     | —      | —     | F1BP<4:0>  | FLTEN0     | —     | —          | —      | —        | —        | —        | —          | F0BP<4:0>  | 0000   |      |
| 0750                          | CFD1FLtCON1                    | 31:16         | FLTEN7        | —     | —      | —     | —          | —     | —      | —     | F7BP<4:0>  | FLTEN6     | —     | —          | —      | —        | —        | —        | —          | F6BP<4:0>  | 0000   |      |
|                               |                                | 15:0          | FLTEN5        | —     | —      | —     | —          | —     | —      | —     | F5BP<4:0>  | FLTEN4     | —     | —          | —      | —        | —        | —        | —          | F4BP<4:0>  | 0000   |      |
| 0760                          | CFD1FLtCON2                    | 31:16         | FLTEN11       | —     | —      | —     | —          | —     | —      | —     | F11BP<4:0> | FLTEN10    | —     | —          | —      | —        | —        | —        | —          | F10BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN9        | —     | —      | —     | —          | —     | —      | —     | F9BP<4:0>  | FLTEN8     | —     | —          | —      | —        | —        | —        | —          | F8BP<4:0>  | 0000   |      |
| 0770                          | CFD1FLtCON3                    | 31:16         | FLTEN15       | —     | —      | —     | —          | —     | —      | —     | F15BP<4:0> | FLTEN14    | —     | —          | —      | —        | —        | —        | —          | F14BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN13       | —     | —      | —     | —          | —     | —      | —     | F13BP<4:0> | FLTEN12    | —     | —          | —      | —        | —        | —        | —          | F12BP<4:0> | 0000   |      |
| 0780                          | CFD1FLtCON4                    | 31:16         | FLTEN19       | —     | —      | —     | —          | —     | —      | —     | F19BP<4:0> | FLTEN18    | —     | —          | —      | —        | —        | —        | —          | F18BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN17       | —     | —      | —     | —          | —     | —      | —     | F17BP<4:0> | FLTEN16    | —     | —          | —      | —        | —        | —        | —          | F16BP<4:0> | 0000   |      |
| 0790                          | CFD1FLtCON5                    | 31:16         | FLTEN23       | —     | —      | —     | —          | —     | —      | —     | F23BP<4:0> | FLTEN22    | —     | —          | —      | —        | —        | —        | —          | F22BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN21       | —     | —      | —     | —          | —     | —      | —     | F21BP<4:0> | FLTEN20    | —     | —          | —      | —        | —        | —        | —          | F20BP<4:0> | 0000   |      |
| 07A0                          | CFD1FLtCON6                    | 31:16         | FLTEN27       | —     | —      | —     | —          | —     | —      | —     | F27BP<4:0> | FLTEN26    | —     | —          | —      | —        | —        | —        | —          | F26BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN25       | —     | —      | —     | —          | —     | —      | —     | F25BP<4:0> | FLTEN24    | —     | —          | —      | —        | —        | —        | —          | F24BP<4:0> | 0000   |      |
| 07B0                          | CFD1FLtCON7                    | 31:16         | FLTEN31       | —     | —      | —     | —          | —     | —      | —     | F31BP<4:0> | FLTEN30    | —     | —          | —      | —        | —        | —        | —          | F28BP<4:0> | 0000   |      |
|                               |                                | 15:0          | FLTEN29       | —     | —      | —     | —          | —     | —      | —     | F29BP<4:0> | FLTEN28    | —     | —          | —      | —        | —        | —        | —          | F28BP<4:0> | 0000   |      |
| 07C0                          | CFD1FLtOBjn<br>(n' = 0)        | 31:16         | —             | EXIDE | SID11  |       |            |       |        |       | EID<17:5>  |            |       |            |        | 0000     |          |          |            |            |        |      |
|                               |                                | 15:0          | EID<4:0>      |       |        |       |            |       |        |       |            | SID<10:0>  |       |            |        |          | 0000     |          |            |            |        |      |
| 07D0                          | CFD1MASKn<br>(n' = 0)          | 31:16         | —             | MIDE  | MSID11 |       |            |       |        |       | MEID<17:5> |            |       |            |        | 0000     |          |          |            |            |        |      |
|                               |                                | 15:0          | MEID<4:0>     |       |        |       |            |       |        |       |            | MSID<10:0> |       |            |        |          | 0000     |          |            |            |        |      |
| 07E0-                         | CFD1FLtOBjn<br>(n' = 1 to 31)  | 31:16         | —             | EXIDE | SID11  |       |            |       |        |       | EID<17:5>  |            |       |            |        | 0000     |          |          |            |            |        |      |
|                               |                                | 15:0          | EID<4:0>      |       |        |       |            |       |        |       |            | SID<10:0>  |       |            |        |          | 0000     |          |            |            |        |      |
| 0BB0                          | CFD1MASKn<br>(n' = 1 to 31)    | 31:16         | —             | MIDE  | MSID11 |       |            |       |        |       | MEID<17:5> |            |       |            |        | 0000     |          |          |            |            |        |      |
|                               |                                | 15:0          | MEID<4:0>     |       |        |       |            |       |        |       |            | MSID<10:0> |       |            |        |          | 0000     |          |            |            |        |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note 1:** The lower order byte of the 32-bit register resides at the low-order address.

**TABLE 24-1: CAN FD PERIPHERAL REGISTER SUMMARY (CONTINUED)**

| Virtual Address<br>(BF88_#) | Register Name | Bit Range | Bits       |            |       |        |       |            |            |          |            |          |            |       |      |          |       | All Resets |      |
|-----------------------------|---------------|-----------|------------|------------|-------|--------|-------|------------|------------|----------|------------|----------|------------|-------|------|----------|-------|------------|------|
|                             |               |           | 31/15      | 30/14      | 29/13 | 28/12  | 27/11 | 26/10      | 25/9       | 24/8     | 23/7       | 22/6     | 21/5       | 20/4  | 19/3 | 18/2     | 17/1  |            | 16/0 |
| 1000                        | CFD2CON       | 31:16     | TXBWS<3:0> |            |       |        | ABAT  | REQOP<2:0> |            |          | OPMOD<2:0> |          |            | TXQEN | STEF | SERR2LOM | ESIGM | RTXAT      | 0498 |
|                             |               | 15:0      | ON         | —          | SIDL  | BRSDIS | BUSY  | WFT<1:0>   | WAKFIL     | CLKSEL0  | PXEDIS     | ISOCRCEN | DNCNT<4:0> |       |      |          | 0760  |            |      |
| 1010                        | CFD2NBTCFG    | 31:16     | BRP<7:0>   |            |       |        |       |            | TSEG1<7:0> |          |            |          |            |       | 003E |          |       |            |      |
|                             |               | 15:0      | —          | TSEG2<6:0> |       |        |       |            | —          | SJW<6:0> |            |          |            |       | 0F0E |          |       |            |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.  
**Note 1:** The lower order byte of the 32-bit register resides at the low-order address.



# PIC32MK GPG/MCJ with CAN FD Family

## 24.2 Control Registers

### REGISTER 24-1: CFD1CON: CAN CONTROL REGISTER(2)

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6        | Bit 29/21/13/5          | Bit 28/20/12/4       | Bit 27/19/11/3      | Bit 26/18/10/2          | Bit 25/17/9/1        | Bit 24/16/8/0         |
|-----------|------------------------|-----------------------|-------------------------|----------------------|---------------------|-------------------------|----------------------|-----------------------|
| 31:24     | R/W-0                  | R/W-0                 | R/W-0                   | R/W-0                | S, HC-0             | R/W-1                   | R/W-0                | R/W-0                 |
|           | TXBWS<3:0>             |                       |                         |                      | ABAT                | REQOP<2:0>              |                      |                       |
| 23:16     | R-1                    | R-0                   | R-0                     | R/W-1                | R/W-1               | R/W-0                   | R/W-0                | R/W-0                 |
|           | OPMOD<2:0>             |                       |                         | TXQEN <sup>(1)</sup> | STEF <sup>(1)</sup> | SERR2LOM <sup>(1)</sup> | ESIGM <sup>(1)</sup> | RTXAT <sup>(1)</sup>  |
| 15:8      | R/W-0                  | R/W-0                 | R/W-0                   | R/W-0                | R-0                 | R/W-1                   | R/W-1                | R/W-1                 |
|           | ON                     | —                     | SIDL <sup>(2)</sup>     | BRSDIS               | BUSY                | WFT<1:0>                |                      | WAKFIL <sup>(1)</sup> |
| 7:0       | R/W-0                  | R/W-1                 | R/W-1                   | R/W-0                | R/W-0               | R/W-0                   | R/W-0                | R/W-0                 |
|           | CLKSEL0 <sup>(1)</sup> | PXEDIS <sup>(1)</sup> | ISOCRCEN <sup>(1)</sup> | DNCNT<4:0>           |                     |                         |                      |                       |

|                   |                  |  |
|-------------------|------------------|--|
| <b>Legend:</b>    | S = Settable bit | HC = Cleared by Hardware                     |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **TXBWS<3:0>**: Transmit Bandwidth Sharing bits

Delay between two consecutive transmissions (in arbitration bit times)

1111-1100 = 4096  
 1011 = 2048  
 1010 = 1024  
 1001 = 512  
 1000 = 256  
 0111 = 128  
 0110 = 64  
 0101 = 32  
 0100 = 16  
 0011 = 8  
 0010 = 4  
 0001 = 2  
 0000 = No delay

bit 27 **ABAT**: Abort All Pending Transmissions bit

1 = Signal all transmit buffers to abort transmission  
 0 = Module will clear this bit when all transmissions aborted

bit 26-24 **REQOP<2:0>**: Request Operation Mode bits

111 = Set Restricted Operation mode  
 110 = Set Normal CAN 2.0 mode; error frames on CAN FD frames  
 101 = Set External Loopback mode  
 100 = Set Configuration mode  
 011 = Set Listen Only mode  
 010 = Set Internal Loopback mode  
 001 = Set Disable mode  
 000 = Set Normal CAN FD mode; supports mixing of Full CAN FD and Classic CAN 2.0 frames

**Note 1:** This bit can only be modified in Configuration mode (OPMOD<2:0> bits = 100).

**Note 2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-1: CFD1CON: CAN CONTROL REGISTER(2)

bit 23-21 **OPMOD<2:0>**: Operation Mode Status bits

- 111 = Module is Restricted Operation mode
- 110 = Module is Normal CAN 2.0 mode; error frames on CAN FD frames
- 101 = Module is in External Loopback mode
- 100 = Module is in Configuration mode
- 011 = Module is in Listen Only mode
- 010 = Module is in Internal Loopback mode
- 001 = Module is in Disable mode
- 000 = Module is in Normal CAN FD mode; supports mixing of Full CAN FD and Classic CAN 2.0 frames

**Note:** In Restricted Operation mode, the node is able to receive data and remote frames, and to acknowledge valid frames, but it does not send data frames, remote frames, active error frames, or overload frames.

bit 20 **TXQEN**: Enable Transmit Queue bit<sup>(1)</sup>

- 1 = Enables Transmit Queue and reserves space in RAM
- 0 = Don't reserve space in RAM for Transmit Queue

**Note:** Changes only in Configuration mode, since it changes the addresses in RAM.

bit 19 **STEF**: Store in Transmit Event FIFO bit<sup>(1)</sup>

- 1 = Save transmitted messages in TEF
- 0 = Don't save transmitted messages in TEF

**Note:** Changes only in Configuration mode, since it changes the addresses in RAM.

bit 18 **SERR2LOM**: Transition to Listen Only Mode on System Error bit<sup>(1)</sup>

- 1 = Transition to Listen Only Mode
- 0 = Transition to Restricted Operation Mode

bit 17 **ESIGM**: Transmit ESI in Gateway Mode bit<sup>(1)</sup>

- 1 = ESI is transmitted as recessive when ESI of message is high or CAN controller error passive
- 0 = ESI reflects error status of CAN controller

bit 16 **RTXAT**: Restrict Retransmission Attempts bit<sup>(1)</sup>

- 1 = Restricted retransmission attempts, use the TXAT<1:0> bit (CiFIFOCONn<22:21>)
- 0 = Unlimited number of retransmission attempts, TXAT<1:0> bit will be ignored

bit 15 **ON**: Enable bit

- 1 = CAN module is enabled
- 0 = CAN module is disabled

bit 14 **Unimplemented**: Read as '0'

bit 13 **SIDL**: Stop-in-Idle Control bit<sup>(2)</sup>

- 1 = Stop module operation in Idle mode
- 0 = Don't stop module operation in Idle mode

bit 12 **BRSDIS**: Bit Rate Switching Disable bit

- 1 = Bit Rate Switching is Disabled, regardless of BRS in the Transmit Message Object
- 0 = Bit Rate Switching depends on BRS in the Transmit Message Object

bit 11 **BUSY**: CAN Module is Busy bit

- 1 = The CAN module is active
- 0 = The CAN module is inactive

bit 10-9 **WFT<1:0>**: Selectable Wake-up Filter Time bits

- 11 = T11FILTER (Typical 600 ns)
- 10 = T10FILTER (Typical 375 ns)
- 01 = T01FILTER (Typical 187 ns)
- 00 = T00FILTER (Typical 100 ns)

**Note 1:** This bit can only be modified in Configuration mode (OPMOD<2:0> bits = 100).

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-1: CFD1CON: CAN CONTROL REGISTER(2)

bit 8 **WAKFIL:** Enable CAN Bus Line Wake-up Filter bit<sup>(1)</sup>

- 1 = Use CAN bus line filter for wake-up
- 0 = CAN bus line filter is not used for wake-up

bit 7 **CLKSEL0:** Module Clock Source Select bit<sup>(1)</sup>

- 1 = REFCLK4 is active when the CAN FD module is enabled
- 0 = SYSCLK is active when the CAN FD module is enabled

**Note:** Max CAN-FD clock source cannot exceed 80Mhz. Recommended CAN-FD clock frequency is 20MHz/40MHz/80MHz. Implementing following steps to set Module Clock frequency of 40MHz while maintaining CPU clock frequency of 120MHz:

- Select module clock source as REFCLK4 by configuring CLKSEL0 bit (CFDxCON<7>) = 1
- Select SYSCLK as the input clock fro REFCLK4 by configure ROSEL bits (REFO4CON<3:0>) = 0
- Set REF4CLK prescaler to 3 by configuring RODIV bits (REFO4CON<30:16>) = 1 and ROTRIM bits (REFO4TRIM<31:23> = 256

bit 6 **PXEDIS:** Protocol Exception Event Detection Disabled bit<sup>(1)</sup>

A recessive "res bit" following a recessive FDF bit is called a Protocol Exception.

- 1 = Protocol Exception is treated as a Form Error.
- 0 = If a Protocol Exception is detected, the CAN FD module will enter the Bus Integrating state.

bit 5 **ISOCRCEN:** Enable ISO CRC in CAN FD Frames bit<sup>(1)</sup>

- 1 = Include Stuff Bit Count in CRC Field and use Non-Zero CRC Initialization Vector
- 0 = Do not include Stuff Bit Count in CRC Field and use CRC Initialization Vector with all zeros

bit 4-0 **DNCNT<4:0>:** Device Net Filter Bit Number bits

- 11111-10011 = Invalid Selection (compare with EID<0:17>)
- 10010 = Compare up to data byte 0<7:0> and Byte 1<7:0> and Byte 2<6> with EID17
- 10001 = Compare up to data byte 0<7:0> and Byte 1<7:0> and Byte 2<7> with EID<0:16>
- 10000 = Compare up to data byte 0<7:0> and data byte 1<7:0> with EID<0:15>
- 01111 = Compare up to data byte 0<7:0> and data byte 1<7:1> with EID<0:14>
- 01110 = Compare up to data byte 0<7:0> and data byte 1<7:2> with EID<0:13>
- 01101 = Compare up to data byte 0<7:0> and data byte 1<7:3> with EID<0:12>
- 01100 = Compare up to data byte 0<7:0> and data byte 1<7:4> with EID<0:11>
- 01011 = Compare up to data byte 0<7:0> and data byte 1<7:5> with EID<0:10>
- 01010 = Compare up to data byte 0<7:0> and data byte 1<7:6> with EID<0:9>
- 01001 = Compare up to data byte 0<7:0> and data byte 1<7> with EID<0:8>
- 01000 = Compare up to data byte 0<7:0> with EID<0:7>
- 00111 = Compare up to data byte 0<7:1> with EID<0:6>
- 00110 = Compare up to data byte 0<7:2> with EID<0:5>
- 00101 = Compare up to data byte 0<7:3> with EID<0:4>
- 00100 = Compare up to data byte 0<7:4> with EID<0:3>
- 00011 = Compare up to data byte 0<7:5> with EID<0:2>
- 00010 = Compare up to data byte 0<7:6> with EID<0:1>
- 00001 = Compare up to data byte 0 bit 7 with EID0
- 00000 = Do not compare data bytes

**Note 1:** This bit can only be modified in Configuration mode (OPMOD<2:0> bits = 100).

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-2: CFD1NBTCFG: NOMINAL BIT TIME CONFIGURATION REGISTER (4)**

| Bit Range  | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| BRP<7:0>   |                |                |                |                |                |                |               |               |
| 23:16      | R/W-0          | R/W-0          | R/W-1          | R/W-1          | R/W-1          | R/W-1          | R/W-1         | R/W-0         |
| TSEG1<7:0> |                |                |                |                |                |                |               |               |
| 15:8       | U-0            | U-0            | U-0            | R/W-1          | R/W-1          | R/W-1          | R/W-1         | R/W-1         |
| TSEG2<6:0> |                |                |                |                |                |                |               |               |
| 7:0        | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-1          | R/W-1          | R/W-1         | R/W-1         |
| SJW<6:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-24 **BRP<7:0>**: Baud Rate Prescaler bits

11111111 =  $TQ = 256/F_{SYS}$   
 .  
 .  
 .  
 00000000 =  $TQ = 1/F_{SYS}$

bit 23-16 **TSEG1<7:0>**: Time Segment 1 bits (Propagation Segment + Phase Segment 1)

1111 1111 = Length is  $256 \times TQ$   
 .  
 .  
 .  
 00000000 = Length is  $1 \times TQ$

bit 15 **Unimplemented**: Read as '0'

bit 14-8 **TSEG2<6:0>**: Time Segment 2 bits (Phase Segment 2)

111111 = Length is  $128 \times TQ$   
 .  
 .  
 .  
 000000 = Length is  $1 \times TQ$

bit 7 **Unimplemented**: Read as '0'

bit 6-0 **SJW<6:0>**: Synchronization Jump Width bits

1111111 = Length is  $128 \times TQ$   
 .  
 .  
 .  
 0000000 = Length is  $1 \times TQ$

- Note 1:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).
- 2:** The following apply to this register:
- $TQ = ((BRP + 1) / FCAN)$
  - Nominal Bit Period =  $(TQ * ((SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$
  - Calc Nominal Bit Rate =  $(1 / \text{Bit Period})$
- OR
- Calc CAN Bit Rate =  $(1 / (((BRP + 1) / FCAN) * (SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$
- 3:** The maximum allowed CAN FD error is  $\leq 1\%$  %Error =  $((\text{Calc CAN bit rate} - \text{Desired CAN bit rate}) / \text{Desired CAN bit rate}) * 100$ .
- 4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-3: CFD1DBTCFG: DATA BIT TIME CONFIGURATION REGISTER (4)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | BRP<7:0>       |                |                |                |                |                |               |               |
| 23:16     | U-0            | U-0            | U-0            | R/W-0          | R/W-1          | R/W-1          | R/W-1         | R/W-0         |
|           | TSEG1<4:0>     |                |                |                |                |                |               |               |
| 15:8      | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-1         | R/W-1         |
|           | TSEG2<3:0>     |                |                |                |                |                |               |               |
| 7:0       | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0          | R/W-1         | R/W-1         |
|           | SJW<3:0>       |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-24 **BRP<7:0>**: Baud Rate Prescaler bits

11111111 =  $T_q = 256/F_{SYS}$

.

.

0000 0000 =  $T_q = 1/F_{SYS}$

bit 23-21 **Unimplemented**: Read as '0'

bit 20-16 **TSEG1<4:0>**: Time Segment 1 bits (Propagation Segment + Phase Segment 1)

11111 = Length is 32 x  $T_q$

.

.

00000 = Length is 1 x  $T_q$

bit 15-12 **Unimplemented**: Read as '0'

bit 11-8 **TSEG2<3:0>**: Time Segment 2 bits (Phase Segment 2)

1111 = Length is 16 x  $T_q$

.

.

0000 = Length is 1 x  $T_q$

bit 7-4 **Unimplemented**: Read as '0'

**Note 1:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21> = 100).

**2:** The following apply to this register:

- $T_q = ((BRP + 1)) / FCAN$

- Nominal Bit Period =  $(T_q * ((SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$

- Calc Nominal Bit Rate =  $(1 / \text{Bit Period})$

OR

- Calc CAN Bit Rate =  $(1 / (((BRP + 1)) / FCAN) * (SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$

**3:** The maximum allowed CAN FD error is  $\leq 1\%$  %Error =  $((\text{Calc CAN bit rate} - \text{Desired CAN bit rate}) / \text{Desired CAN bit rate}) * 100$ .

**4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-3: CFD1DBTCFG: DATA BIT TIME CONFIGURATION REGISTER (CONTINUED)(4)

bit 3-0 **SJW<3:0>**: Synchronization Jump Width bits

1111 = Length is 16 x T<sub>Q</sub>

.

.

0000 = Length is 1 x T<sub>Q</sub>

**Note 1:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFD<sub>x</sub>CON<23:21>) = 100).

**2:** The following apply to this register:

- $T_Q = ((BRP + 1) / FCAN)$
- Nominal Bit Period =  $(T_Q * ((SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$
- Calc Nominal Bit Rate =  $(1 / \text{Bit Period})$

OR

- Calc CAN Bit Rate =  $(1 / (((BRP + 1) / FCAN) * (SYNC + (TSEG1 + 1) + (TSEG2 + 1))))$

**3:** The maximum allowed CAN FD error is  $\leq 1\%$  %Error =  $((\text{Calc CAN bit rate} - \text{Desired CAN bit rate}) / \text{Desired CAN bit rate}) * 100$ .

**4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-4: CFD1TDC: TRANSMITTER DELAY COMPENSATION REGISTER (1,2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | R/W-0         |
|           | —              | —              | —              | —              | —              | —              | EDGFLTEN      | SID11EN       |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-1         | R/W-0         |
|           | —              | —              | —              | —              | —              | —              | TDCMOD<1:0>   |               |
| 15:8      | U-0            | R/W-0          | R/W-0          | R/W-1          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | TDCO<6:0>      |                |                |                |                |               |               |
| 7:0       | U-0            | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | TDCV<5:0>      |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31-24 **Unimplemented:** Read as '0'
- bit 25 **EDGFLTEN:** Enable Edge Filtering during Bus Integration state bit  
 1 = Edge Filtering enabled, according to ISO11898-1:2015  
 0 = Edge Filtering disabled
- bit 24 **SID11EN:** Enable 12-Bit SID in CAN FD Base Format Messages bit  
 1 = RRS is used as SID11 in CAN FD base format messages: SID<11:0> = {SID<10:0>, SID11}  
 0 = Do not use RRS; SID<10:0> according to ISO11898-1:2015
- bit 23-18 **Unimplemented:** Read as '0'
- bit 17-16 **TDCMOD<1:0>:** Transmitter Delay Compensation Mode bits  
 Secondary Sample Point (SSP).  
 10 = Auto; measure delay and add CFDxDBTCFG.TSEG1; add TDCO  
 11 = Auto; measure delay and add CFDxDBTCFG.TSEG1; add TDCO  
 01 = Manual; Do not measure, use TDCV plus TDCO from the register  
 00 = Disable
- bit 15 **Unimplemented:** Read as '0'
- bit 14-8 **TDCO<6:0>:** Transmitter Delay Compensation Offset bits  
 Secondary Sample Point (SSP). Two's complement; offset can be positive, zero, or negative.  
 11111111 = -64 x SYSCLK  
 .  
 .  
 .  
 01111111 = 63 x SYSCLK  
 .  
 .  
 .  
 00000000 = 0 x SYSCLK
- bit 7-6 **Unimplemented:** Read as '0'

**Note 1:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).  
**Note 2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-4: CFD1TDC: TRANSMITTER DELAY COMPENSATION REGISTER (1,2)

bit 5-0 **TDCV<5:0>**: Transmitter Delay Compensation Value bits; Secondary Sample Point (SSP)

111111 = 63 x SYSCLK

.

.

000000 = 0 x SYSCLK

**Note 1:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**Note 2:** Not available on PIC32MK GPG variants.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-5: CFD1TBC: CAN TIME BASE COUNTER REGISTER(1,2)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TBC<31:24>     |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TBC<23:16>     |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TBC<15:8>      |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TBC<7:0>       |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **TBC<31:0>**: CAN Base Counter bits

This is a free running 32-bit time stamp timer that increments every SYSCLK clocks (TBCPRE<9:0> (CFDxTSCON<9:0>)) when the TTBCEN bit (CFDxTSCON<8>) is set.

**Note 1:** To save power, the TBC will be stopped and reset when the TBCEN bit (CFDxTSCON<16>) = 0 to save power.

**2:** The TBC prescaler count will be reset on any write to the CFDxTBC register (TBCPRE<9:0> (CFDxTSCON<9:0>)) will be unaffected).

**Note 1:** The timer/counter increments on SYSCLK and rolls over to zero.

There are two main time stamping registers:

- CFDxTBC: Time Base Counter, 32-bit

- CFDxTSCON: Time Stamp Control register

Time Stamp Event selectable:

- Classic CAN Frame: SOF versus EOF

- CAN FD Frame: After SOF/FDF versus EOF

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-6: CFD1TSCON: CAN TIME STAMP CONTROL REGISTER (1)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0          | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | —              | TSRES          | TSEOF         | TBCEN         |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | R/W-0         | R/W-0         |
|           | —              | —              | —              | —              | —              | —              | TBCPRE<9:8>   |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TBCPRE<7:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-19 **Unimplemented:** Read as '0'

bit 18 **TSRES:** Time Stamp Resolution bit  
 FD Frames only.  
 1 = At sample point of the bit following the FDF bit  
 0 = At sample point of SOF

bit 17 **TSEOF:** Time Stamp EOF bit  
 1 = Time Stamp when frame is taken valid (11898-1 10.7):  
 • RX no error until last but one bit of EOF)  
 • TX no error until the end of EOF  
 0 = Time Stamp at “beginning” of Frame:  
 • Classical Frame: At sample point of SOF  
 • FD Frame: See the TSRES bit

bit 16 **TBCEN:** Time Base Counter Enable bit  
 1 = Enable TBC  
 0 = Stop and reset TBC

bit 15-10 **Unimplemented:** Read as '0'

bit 9-0 **TBCPRE<9:0>:** CAN Time Base Counter Prescaler bits  
 111111111 = TBC increments every 1024 SYSCLK clock cycles  
 .  
 .  
 .  
 000000000 = TBC increments every 1 SYSCLK clock cycle

**Note 1:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-7: CFD1VEC: INTERRUPT CODE REGISTER (1,2)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6             | Bit 29/21/13/5 | Bit 28/20/12/4             | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------------------|----------------|----------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | R-1                        | R-0            | R-0                        | R-0            | R-0            | R-0           | R-0           |
|           | —              | RXCODE<6:0> <sup>(1)</sup> |                |                            |                |                |               |               |
| 23:16     | U-0            | R-1                        | R-0            | R-0                        | R-0            | R-0            | R-0           | R-0           |
|           | —              | TXCODE<6:0> <sup>(1)</sup> |                |                            |                |                |               |               |
| 15:8      | U-0            | U-0                        | U-0            | R-0                        | R-0            | R-0            | R-0           | R-0           |
|           | —              | —                          | —              | FILHIT<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | U-0            | R-1                        | R-0            | R-0                        | R-0            | R-0            | R-0           | R-0           |
|           | —              | ICODE<6:0> <sup>(1)</sup>  |                |                            |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **Unimplemented:** Read as '0'

bit 30-24 **RXCODE<6:0>:** Receive Interrupt Flag Code bits<sup>(1)</sup>

1111111 = Reserved

.

.

.

1000001 = Reserved

1000000 = No interrupt

0111111 = Reserved

.

.

.

0100000 = Reserved

0011111 = FIFO 31 Interrupt (RFIF<31> set)

.

.

.

0000010 = FIFO 2 Interrupt (RFIF<2> set)

0000001 = FIFO 1 Interrupt (RFIF<1> set)

0000000 = Reserved. FIFO 0 can't receive.

bit 23 **Unimplemented:** Read as '0'

**Note 1:** If multiple interrupts are pending, the interrupt with the highest number will be indicated.

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-7: CFD1VEC: INTERRUPT CODE REGISTER (1,2)

bit 22-16 **TXCODE<6:0>**: Transmit Interrupt Flag Code bits <sup>(1)</sup>

1111111 = Reserved  
.  
.  
1000001 = Reserved  
1000000 = No interrupt  
0111111 = Reserved  
.  
.  
0100000 = Reserved  
0011111 = FIFO 31 interrupt (TFIF<31> bit is set)  
.  
.  
0000001 = FIFO 1 interrupt (TFIF<1> bit is set)  
0000000 = FIFO 0 interrupt (TFIF<0> bit is set)

bit 15-13 **Unimplemented**: Read as '0'

bit 12-8 **FILHIT<4:0>**: Filter Hit Number bits<sup>(1)</sup>

11111 = Filter 31  
11110 = Filter 30  
.  
.  
00001 = Filter 1  
00000 = Filter 0

bit 7 **Unimplemented**: Read as '0'

bit 6-0 **ICODE<6:0>**: Interrupt Flag Code bits<sup>(1)</sup>

1111111 = Reserved  
.  
.  
1001011 = Reserved  
1001010 = Transmit attempt interrupt (any bit in the CFDxTXATIF register is set)  
1001001 = Transmit event FIFO interrupt (any bit in the CFDxTEFIF register is set)  
1001000 = Invalid message occurred (IVMIF/IE)  
1000111 = CAN module mode change Occurred (MODIF/IE)  
1000110 = CAN Timer Overflow (CTMRIF/IE)  
1000101 = RX/TX MAB overflow/underflow (RX: message received before previous message was saved to memory; TX: cannot feed TX MAB fast enough to transmit consistent data.) (SERRIF/IE)  
1000100 = Address error interrupt (illegal FIFO address presented to system) (SERRIF/IE)  
1000011 = Receive FIFO overflow interrupt (any bit in CFDxRXOVIF set)  
1000010 = Wake-up interrupt (WAKIF/WAKIE)  
1000001 = Error interrupt (CERRIF/IE)  
1000000 = No interrupt  
0111111 = Reserved  
0100000 = Reserved  
0011111 = FIFO 31 interrupt (TFIF<31> or RFIF<31> set)  
.  
.  
0000001 = FIFO 1 interrupt (TFIF<1> or RFIF<1> set)  
0000000 = FIFO 0 interrupt (TFIF<0> set)

**Note 1:** If multiple interrupts are pending, the interrupt with the highest number will be indicated.

**Note 2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-8: CFD1INT: INTERRUPT REGISTER (1,2)**

| Bit Range | Bit 31/23/15/7       | Bit 30/22/14/6       | Bit 29/21/13/5        | Bit 28/20/12/4        | Bit 27/19/11/3       | Bit 26/18/10/2       | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------|----------------------|-----------------------|-----------------------|----------------------|----------------------|---------------|---------------|
| 31:24     | R/W-0                | R/W-0                | R/W-0                 | R/W-0                 | R/W-0                | R/W-0                | U-0           | U-0           |
|           | IVMIE                | WAKIE                | CERRIE                | SERRIE                | RXOVIE               | TXATIE               | —             | —             |
| 23:16     | U-0                  | U-0                  | U-0                   | R/W-0                 | R/W-0                | R/W-0                | R/W-0         | R/W-0         |
|           | —                    | —                    | —                     | TEFIE                 | MODIE                | TBCIE                | RXIE          | TXIE          |
| 15:8      | HS/C-0               | HS/C-0               | HS/C-0                | HS/C-0                | R-0                  | R-0                  | U-0           | U-0           |
|           | IVMIF <sup>(1)</sup> | WAKIF <sup>(1)</sup> | CERRIF <sup>(1)</sup> | SERRIF <sup>(1)</sup> | RXOVIF               | TXATIF               | —             | —             |
| 7:0       | U-0                  | U-0                  | U-0                   | R-0                   | HS/C-0               | HS/C-0               | R-0           | R-0           |
|           | —                    | —                    | —                     | TEFIF                 | MODIF <sup>(1)</sup> | TBCIF <sup>(1)</sup> | RXIF          | TXIF          |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 31 **IVMIE:** Invalid Message Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 30 **WAKIE:** Bus Wake Up Activity Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 29 **CERRIE:** CAN Bus Error Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 28 **SERRIE:** System Error Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 27 **RXOVIE:** Receive Buffer Overflow Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 26 **TXATIE:** Transmit Attempt Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 25-21 **Unimplemented:** Read as '0'
- bit 20 **TEFIE:** Transmit Event FIFO Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 19 **MODIE:** Mode Change Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 18 **TBCIE:** CAN Timer Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled

**Note 1:** Flags are set by hardware and are cleared by the user application.

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-8: CFD1INT: INTERRUPT REGISTER (1,2)

- bit 17 **RXIE:** Receive Object Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 16 **TXIE:** Transmit Object Interrupt Enable bit  
1 = Interrupts are enabled  
0 = Interrupts are disabled
- bit 15 **IVMIF:** Invalid Message Interrupt Flag bit<sup>(1)</sup>  
1 = An invalid message interrupt is pending  
0 = No invalid message interrupts are pending
- bit 14 **WAKIF:** Bus Wake Up Activity Interrupt Flag bit<sup>(1)</sup>  
1 = A wake-up interrupt is pending  
0 = No wake-up interrupts are pending
- bit 13 **CERRIF:** CAN Bus Error Interrupt Flag bit<sup>(1)</sup>  
1 = A CAN bus error interrupt is pending  
0 = No CAN bus error interrupts are pending
- bit 12 **SERRIF:** System Error Interrupt Flag bit<sup>(1)</sup>  
1 = A system error occurred (collision on dual-port RAM)  
0 = No system error occurred
- bit 11 **RXOVIF:** Receive Object Overflow Interrupt Flag bit  
1 = Receive object overflow occurred  
0 = No receive object overflow has occurred
- bit 10 **TXATIF:** Transmit Attempt Interrupt Flag bit  
1 = A transmit interrupt is pending in one or more of the designated 32 FIFOs denoted by which of the 32 bits in the CFDTXATIF register are set  
0 = No transmit FIFO interrupts are pending
- bit 9-5 **Unimplemented:** Read as '0'
- bit 4 **TEFIF:** Transmit Event FIFO Interrupt Flag bit  
1 = Receive buffer overflow occurred  
0 = No receive buffer overflow has occurred
- bit 3 **MODIF:** CAN Mode Change Interrupt Flag bit<sup>(1)</sup>  
1 = CAN module mode change occurred (OPMOD has changed to reflect the REQOP<2:0> bits)  
0 = No mode change occurred
- bit 2 **TBCIF:** CAN Timer Overflow Interrupt Flag bit<sup>(1)</sup>  
1 = TBC has overflowed  
0 = TBC did not overflow
- bit 1 **RXIF:** Receive Object Interrupt Flag bit  
1 = Receive object interrupt pending  
0 = No receive object interrupts pending
- bit 0 **TXIF:** Transmit Object Interrupt Flag bit  
1 = Transmit object interrupt pending  
0 = No transmit object interrupts pending

**Note 1:** Flags are set by hardware and are cleared by the user application.

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-9: CFD1RXIF: RECEIVE INTERRUPT STATUS REGISTER (2)**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFIF<31:24> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 23:16     | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFIF<23:16> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 15:8      | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFIF<15:8> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 7:0       | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | U-0           |
|           | RFIF<7:1> <sup>(1)</sup>   |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-1 **RFIF<31:1>**: Receive FIFO Interrupt Pending bits<sup>(1)</sup>  
 1 = One or more enabled receive FIFO interrupts are pending  
 0 = No enabled receive FIFO interrupts are pending

bit 0 **Unimplemented**: Read as '0'

- Note 1:** RFIF = 'or' of enabled RXFIFO flags; (flags need to be cleared in FIFO register).  
**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-10: CFD1RXOVIF: RECEIVE OVERFLOW INTERRUPT STATUS REGISTER (2)

| Bit Range | Bit 31/23/15/7               | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|------------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-0                          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFOVIF<31:24> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 23:16     | R-0                          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFOVIF<23:16> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 15:8      | R-0                          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RFOVIF<15:8> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 7:0       | R-0                          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | U-0           |
|           | RFOVIF<7:1> <sup>(1)</sup>   |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-1 **RFOVIF<31:1>**: Receive FIFO Overflow Interrupt Pending bits<sup>(1)</sup>

1 = Interrupt is pending

0 = Interrupt not pending

bit 0 **Unimplemented**: Read as '0'

**Note 1:** RVOVIF (flag need to be cleared in FIFO register).

**2:** Not available on PIC32MK GPG variants.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-11: CFD1TXIF: TRANSMIT INTERRUPT STATUS REGISTER (3)**

| Bit Range | Bit 31/23/15/7              | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-0                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | TFIF<31:24> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 23:16     | R-0                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | TFIF<23:16> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 15:8      | R-0                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | TFIF<15:8> <sup>(1)</sup>   |                |                |                |                |                |               |               |
| 7:0       | R-0                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | TFIF<7:0> <sup>(1, 2)</sup> |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **TFIF<31:0>**: Transmit FIFO/Transmit Queue<sup>(2)</sup> Interrupt Pending bits<sup>(1)</sup>

1 = One or more enabled transmit FIFO/Transmit Queue interrupts are pending

0 = No enabled transmit FIFO/Transmit Queue interrupt are pending

**Note 1:** TFIF = 'or' of the enabled TXFIFO flags; (flags need to be cleared in FIFO register).

**2:** TFIF<0> is for the Transmit Queue.

**3:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-12: CFD1TXATIF: TRANSMIT ATTEMPT INTERRUPT STATUS REGISTER (3)

| Bit Range                     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-------------------------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| TFATIF<31:24> <sup>(1)</sup>  |                |                |                |                |                |                |               |               |
| 23:16                         | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| TFATIF<23:16> <sup>(1)</sup>  |                |                |                |                |                |                |               |               |
| 15:8                          | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| TFATIF<15:8> <sup>(1)</sup>   |                |                |                |                |                |                |               |               |
| 7:0                           | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
| TFATIF<7:0> <sup>(1, 2)</sup> |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **TFATIF<31:0>**: Transmit FIFO/Transmit Queue<sup>(2)</sup> Attempt Interrupt Pending bits<sup>(1)</sup>

1 = A transmit interrupt is pending in one or more of the designated 32 FIFOs denoted by which of the 32 bits in the CFDxTXATIF register are set

0 = No transmit FIFO interrupts are pending

**Note 1:** TFATIF = 'or' of the enabled TXFIFO flags; (flags need to be cleared in FIFO register).

**2:** TFATIF<0> is for the Transmit Queue.

**3:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-13: CFD1TXREQ: TRANSMIT REQUEST REGISTER (2)

| Bit Range | Bit 31/23/15/7              | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | S, HC-0                     | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0       | S, HC-0       |
|           | TXREQ<31:24> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 23:16     | S, HC-0                     | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0       | S, HC-0       |
|           | TXREQ<23:16> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 15:8      | S, HC-0                     | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0       | S, HC-0       |
|           | TXREQ<15:8> <sup>(1)</sup>  |                |                |                |                |                |               |               |
| 7:0       | S, HC-0                     | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0        | S, HC-0       | S, HC-0       |
|           | TXREQ<7:0> <sup>(1)</sup>   |                |                |                |                |                |               |               |

|                   |                  |                      |                                    |
|-------------------|------------------|----------------------|------------------------------------|
| <b>Legend:</b>    | S = Settable bit | W = Writable bit     | U = Unimplemented bit, read as '0' |
| R = Readable bit  |                  |                      |                                    |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown                 |

bit 31-1 **TXREQ<31:1>**: Message Send Request bits<sup>(1)</sup>

TXEN = 1: (Object configured as a Transmit Object)

Setting this bit to '1' requests sending a message.

The bit will automatically clear when the message(s) queued in the object is (are) successfully sent and cannot be used for aborting a transmission.

TXEN = 0: (Object configured as a Receive Object)

This bit has no effect.

bit 0 **TXREQ<0>**: Transmit Queue Message Send Request bit<sup>(1)</sup>

Setting this bit to '1' requests sending a message.

The bit will automatically clear when the message(s) queued in the object is (are) successfully sent and cannot be used for aborting a transmission.

**Note 1:** The TXREQ<31:0> bits are ignored in Listen Only mode (REQOP<2:0> bits (CFDxCON<26:24>) = 0'b011).

**2:** Not available on Pic32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-14: CFD1FIFOBA: MESSAGE MEMORY BASE ADDRESS REGISTER(1,2,3)**

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| FIFOBA<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| FIFOBA<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | SR/W-0         | R/W-0          | R/W-0         | R/W-0         |
| FIFOBA<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | U-0           | U-0           |
| FIFOBA<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **FIFOBA<31:0>**: Message Memory Base Address bits

Defines the base address for Transmit Event FIFO followed by the message objects.

**Note 1:** Bits<1:0> are forced to '0' to be word aligned.

**2:** This register can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**3:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-15: CFD1TXQCON: TRANSMIT QUEUE CONTROL REGISTER(3)**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2        | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|----------------|---------------------------|----------------|-----------------------|---------------|---------------|
| 31:24     | R/W-0                      | R/W-0          | R/W-0          | R/W-0                     | R/W-0          | R/W-0                 | R/W-0         | R/W-0         |
|           | PLSIZE<2:0> <sup>(1)</sup> |                |                | FSIZE<4:0> <sup>(1)</sup> |                |                       |               |               |
| 23:16     | U-0                        | R/W-1          | R/W-1          | R/W-0                     | R/W-0          | R/W-0                 | R/W-0         | R/W-0         |
|           | —                          | TXAT<1:0>      |                | TXPRI<4:0>                |                |                       |               |               |
| 15:8      | U-0                        | U-0            | U-0            | U-0                       | U-0            | S, HC-1               | R/W, HC-0     | S, HC-0       |
|           | —                          | —              | —              | —                         | —              | FRESET <sup>(2)</sup> | TXREQ         | UINC          |
| 7:0       | R-1                        | U-0            | U-0            | R/W-0                     | U-0            | R/W-0                 | U-0           | R/W-0         |
|           | TXEN                       | —              | —              | TXATIE                    | —              | TXQEIE                | —             | TXQNIE        |

**Legend:**

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-29 **PLSIZE<2:0>**: Payload Size bits<sup>(1)</sup>

- 111 = 64 data bytes
- 110 = 48 data bytes
- 101 = 32 data bytes
- 100 = 24 data bytes
- 011 = 20 data bytes
- 010 = 16 data bytes
- 001 = 12 data bytes
- 000 = 8 data bytes

bit 28-24 **FSIZE<4:0>**: FIFO Size bits<sup>(1)</sup>

- 11111 = FIFO is 32 Messages deep
- .
- .
- 00010 = FIFO is 3 Messages deep
- 00001 = FIFO is 2 Messages deep
- 00000 = FIFO is 1 Message deep

bit 23 **Unimplemented**: Read as '0'

bit 22-21 **TXAT<1:0>**: Retransmission Attempts bits

This feature is enabled when the RTXAT bit (CFDxCON<8>) is set.

- 11 = Unlimited number of retransmission attempts
- 10 = Unlimited number of retransmission attempts
- 01 = Three retransmission attempts
- 00 = Disable retransmission attempts

**Note:** The user application must be able to change these bits in Normal mode. This can be used to go back on the bus after bus off to check if transmission works again.

**Note 1:** This bit can only be modified in Configuration mode ((OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**2:** This bit is set while in Configuration mode and is automatically cleared in Normal mode.

**3:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-15: CFD1TXQCON: TRANSMIT QUEUE CONTROL REGISTER(3)

bit 20-16 **TXPRI<4:0>**: Message Transmit Priority bits (**EXPANDED**)

11111 = Highest Message Priority

·  
·  
·

00000 = Lowest Message Priority

bit 15-11 **Unimplemented**: Read as '0'

bit 10 **FRESET**: FIFO Reset bit<sup>(2)</sup>

1 = FIFO will be reset when bit is set; cleared by hardware when FIFO is reset. The user application should poll whether this bit is clear before taking any action.

0 = No effect

bit 9 **TXREQ**: Message Send Request bit

1 = Requests sending a message; the bit will automatically clear when all the messages queued in the Transmit Queue are successfully sent

0 = Clearing the bit to '0' while set ('1') will request a message abort.

bit 8 **UINC**: Increment Head/Tail bit

When this bit is set, the FIFO head will increment by a single message.

bit 7 **TXEN**: TX Enable

1 = Transmit Message Queue. This bit always reads as '1'.

bit 6-5 **Unimplemented**: Read as '0'

bit 4 **TXATIE**: Transmit Attempts Exhausted Interrupt Enable bit

1 = Enable interrupt

0 = Disable interrupt

bit 3 **Unimplemented**: Read as '0'

bit 2 **TXQEIE**: Transmit Queue Empty Interrupt Enable bit

1 = Interrupt enabled for Transmit Queue empty

0 = Interrupt disabled for Transmit Queue empty

bit 1 **Unimplemented**: Read as '0'

bit 0 **TXQNie**: Transmit Queue Not Full Interrupt Enable bit

1 = Interrupt enabled for Transmit Queue not full

0 = Interrupt disabled for Transmit Queue not full

**Note 1:** This bit can only be modified in Configuration mode ((OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**2:** This bit is set while in Configuration mode and is automatically cleared in Normal mode.

**3:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-16: CFD1TXQSTA: TRANSMIT QUEUE STATUS REGISTER (4)**

| Bit Range | Bit 31/23/15/7         | Bit 30/22/14/6          | Bit 29/21/13/5         | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|------------------------|-------------------------|------------------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                    | U-0                     | U-0                    | U-0                       | U-0            | U-0            | U-0           | U-0           |
|           | —                      | —                       | —                      | —                         | —              | —              | —             | —             |
| 23:16     | U-0                    | U-0                     | U-0                    | U-0                       | U-0            | U-0            | U-0           | U-0           |
|           | —                      | —                       | —                      | —                         | —              | —              | —             | —             |
| 15:8      | U-0                    | U-0                     | U-0                    | R-0                       | R-0            | R-0            | R-0           | R-0           |
|           | —                      | —                       | —                      | TXQCI<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R-0                    | R-0                     | R-0                    | HS/C-0                    | U-0            | R-0            | U-0           | R-0           |
|           | TXABT <sup>(2,3)</sup> | TXLARB <sup>(2,3)</sup> | TXERR <sup>(2,3)</sup> | TXATIF                    | —              | TXQEIF         | —             | TXQNIF        |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-13 **Unimplemented:** Read as '0'

bit 12-8 **TXQCI<4:0>:** Transmit Queue Message Index bits<sup>(1)</sup>

A read of this register will return an index to the message that the FIFO will next attempt to transmit.

bit 7 **TXABT:** Message Aborted Status bit<sup>(2,3)</sup>

1 = Message was aborted  
 0 = Message completed successfully

bit 6 **TXLARB:** Message Lost Arbitration Status bit<sup>(2,3)</sup>

1 = Message lost arbitration while being sent  
 0 = Message did not loose arbitration while being sent

bit 5 **TXERR:** Error Detected During Transmission bit<sup>(2,3)</sup>

1 = A bus error occurred while the message was being sent  
 0 = A bus error did not occur while the message was being sent

bit 4 **TXATIF:** Transmit Attempts Exhausted Interrupt Pending bit

1 = Interrupt pending  
 0 = Interrupt Not pending

bit 3 **Unimplemented:** Read as '0'

bit 2 **TXQEIF:** Transmit Queue Empty Interrupt Flag bit

1 = Transmit Queue is empty  
 0 = Transmit Queue is not empty, at least one message queued to be transmitted

bit 1 **Unimplemented:** Read as '0'

bit 0 **TXQNIF:** Transmit Queue Not Full Interrupt Flag bit

1 = Transmit Queue is not full  
 0 = Transmit Queue is full

**Note 1:** TXQCI<4:0> gives a zero-indexed value to the message in the Transmit Queue. If the Transmit Queue is four messages deep (FSIZE<4:0> bits (CFDxFIFOCONn<28:24>) = 5'h03), TXQCI<4:0> will take on a value of 0 to 3 depending on the state of the Transmit Queue.

**2:** This bit is reset on any read of this register or when the Transmit Queue is reset.

**3:** This bit is updated when a message completes (or aborts) or when the Transmit Queue is reset.

**4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-17: CFD1FIFOCONn: FIFO CONTROL REGISTER ('n' = 1-31) (4)**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5        | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|-----------------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0                      | R/W-0          | R/W-0                 | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | PLSIZE<2:0> <sup>(1)</sup> |                |                       | FSIZE<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | U-0                        | R/W-1          | R/W-1                 | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —                          | TXAT<1:0>      |                       | TXPRI<4:0>                |                |                |               |               |
| 15:8      | U-0                        | U-0            | U-0                   | U-0                       | U-0            | S, HC-1        | R/W, HC-0     | S, HC-0       |
|           | —                          | —              | —                     | —                         | —              | FRESET         | TXREQ         | UINC          |
| 7:0       | R/W-0                      | R/W-0          | R/W-0                 | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TXEN <sup>(1)</sup>        | RTREN          | RXTSEN <sup>(1)</sup> | TXATIE                    | RXOVIE         | TFERFFIE       | TFHRFHIE      | TFNRFNIE      |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-29 **PLSIZE<2:0>**: Payload Size bits<sup>(1)</sup>

- 111 = 64 data bytes
- 110 = 48 data bytes
- 101 = 32 data bytes
- 100 = 24 data bytes
- 011 = 20 data bytes
- 010 = 16 data bytes
- 001 = 12 data bytes
- 000 = 8 data bytes

bit 28-24 **FSIZE<4:0>**: FIFO Size bits<sup>(1)</sup>

- 11111 = FIFO is 32 Messages deep
- .
- .
- .
- 00010 = FIFO is 3 Messages deep
- 00001 = FIFO is 2 Messages deep
- 00000 = FIFO is 1 Message deep

bit 23 **Unimplemented**: Read as '0'

bit 22-21 **TXAT<1:0>**: Retransmission Attempts bits

This feature is enabled when the RTXAT bit (CFDxCON<8>) is set.

- 11 = Unlimited number of retransmission attempts
- 10 = Unlimited number of retransmission attempts
- 01 = Three retransmission attempts
- 00 = Disable retransmission attempts

**Note:** Application must be able to change this in Normal mode. This can be used to go back on the bus after bus off to check if transmission works again

- Note 1:** This bit can only be modified in Configuration mode ((OPMOD<2:0> bits (CFDxCON<23:21>) = 100).
- Note 2:** FRESET is set while in Configuration mode and is automatically cleared in Normal mode.
- Note 3:** This bit is updated when a message completes (or aborts) or when the FIFO is reset.
- Note 4:** Not available on PIC32MK GPG variants.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-17: CFD1FIFOCONn: FIFO CONTROL REGISTER ('n' = 1-31) (4) (CONTINUED)

bit 20-16 **TXPRI<4:0>**: Message Transmit Priority bits (**EXPANDED**)

11111 = Highest Message Priority

.

.

.

00000 = Lowest Message Priority

bit 15-11 **Unimplemented**: Read as '0'

bit 10 **FRESET**: FIFO Reset bit<sup>(2)</sup>

1 = FIFO will be reset when bit is set; cleared by hardware when FIFO is reset. User should poll whether this bit is clear before taking any action

0 = No effect

bit 9 **TXREQ**: Message Send Request bit<sup>(3)</sup>

TXEN = 1: (FIFO configured as a Transmit FIFO)

1 = Requests sending a message; the bit will automatically clear when all the messages queued in the FIFO are successfully sent

0 = Clearing the bit to '0' while set ('1') will request a message abort.

TXEN = 0: (FIFO configured as a Receive FIFO)

This bit has no effect.

bit 8 **UINC**: Increment Head / Tail bit

TXEN = 1: (FIFO configured as a Transmit FIFO)

When this bit is set, the FIFO head will increment by a single message

TXEN = 0: (FIFO configured as a Receive FIFO)

When this bit is set, the FIFO tail will increment by a single message

bit 7 **TXEN**: TX / RX Buffer Selection bit<sup>(1)</sup>

1 = Transmit Message Object

0 = Receive Message Object

bit 6 **RTREN**: Auto RTR Enable bit

1 = When a remote transmit is received, TXREQ will be set.

0 = When a remote transmit is received, TXREQ will be unaffected.

bit 5 **RXTSEN**: Received Message Time Stamp Enable bit<sup>(1)</sup>

1 = Capture time stamp in received message object in RAM.

0 = Don't capture time stamp.

**Note**: Change only in Configuration mode, since it is used for address calculation.

bit 4 **TXATIE**: Transmit Attempts Exhausted Interrupt Enable bit

1 = Enable interrupt

0 = Disable interrupt

bit 3 **RXOVIE**: Overflow Interrupt Enable bit

1 = Interrupt enabled for overflow event

0 = Interrupt disabled for overflow event

**Note 1**: This bit can only be modified in Configuration mode ((OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**2**: FRESET is set while in Configuration mode and is automatically cleared in Normal mode.

**3**: This bit is updated when a message completes (or aborts) or when the FIFO is reset.

**4**: Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-17: CFD1FIFOCONn: FIFO CONTROL REGISTER ('n' = 1-31) (4) (CONTINUED)

- bit 2     **TFERFFIE**: Transmit/Receive FIFO Empty/Full Interrupt Enable bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Empty Interrupt Enable  
          1 = Interrupt enabled for FIFO empty  
          0 = Interrupt disabled for FIFO empty  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Full Interrupt Enable  
          1 = Interrupt enabled for FIFO full  
          0 = Interrupt disabled for FIFO full
- bit 1     **TFHRFHIE**: Transmit/Receive FIFO Half Empty/Half Full Interrupt Enable bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Half Empty Interrupt Enable  
          1 = Interrupt enabled for FIFO half empty  
          0 = Interrupt disabled for FIFO half empty  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Half Full Interrupt Enable  
          1 = Interrupt enabled for FIFO half full  
          0 = Interrupt disabled for FIFO half full
- bit 0     **TFNRFNIE**: Transmit/Receive FIFO Not Full/Not Empty Interrupt Enable bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Not Full Interrupt Enable  
          1 = Interrupt enabled for FIFO not full  
          0 = Interrupt disabled for FIFO not full  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Not Empty Interrupt Enable  
          1 = Interrupt enabled for FIFO not empty  
          0 = Interrupt disabled for FIFO not empty

- Note 1:** This bit can only be modified in Configuration mode ((OPMOD<2:0> bits (CFDxCON<23:21>) = 100).
- 2:** FRESET is set while in Configuration mode and is automatically cleared in Normal mode.
- 3:** This bit is updated when a message completes (or aborts) or when the FIFO is reset.
- 4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-18: CFD1TXQSTAn: TRANSMIT QUEUE STATUS REGISTER ('n' = 1-31) (4)**

| Bit Range | Bit 31/23/15/7                | Bit 30/22/14/6                 | Bit 29/21/13/5                | Bit 28/20/12/4             | Bit 27/19/11/3   | Bit 26/18/10/2  | Bit 25/17/9/1   | Bit 24/16/8/0   |
|-----------|-------------------------------|--------------------------------|-------------------------------|----------------------------|------------------|-----------------|-----------------|-----------------|
| 31:24     | U-0<br>—                      | U-0<br>—                       | U-0<br>—                      | U-0<br>—                   | U-0<br>—         | U-0<br>—        | U-0<br>—        | U-0<br>—        |
| 23:16     | U-0<br>—                      | U-0<br>—                       | U-0<br>—                      | U-0<br>—                   | U-0<br>—         | U-0<br>—        | U-0<br>—        | U-0<br>—        |
| 15:8      | U-0<br>—                      | U-0<br>—                       | U-0<br>—                      | FIFOCI<4:0> <sup>(1)</sup> |                  |                 |                 |                 |
| 7:0       | R-0<br>TXABT <sup>(2,3)</sup> | R-0<br>TXLARB <sup>(2,3)</sup> | R-0<br>TXERR <sup>(2,3)</sup> | HS/C-0<br>TXATIF           | HS/C-0<br>RXOVIF | R-0<br>TFERFFIF | R-0<br>TFHRFHIF | R-0<br>TFNRFNIF |

**Legend:**

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31-13 **Unimplemented:** Read as '0'

bit 12-8 **FIFOCI<4:0>:** FIFO Message Index bits<sup>(1)</sup>

TXEN = 1: (FIFO configured as a Transmit Buffer)

A read of this register will return an index to the message that the FIFO will next attempt to transmit.

TXEN = 0: (FIFO configured as a Receive Buffer)

A read of this register will return an index to the message that the FIFO will use to save the next message.

bit 7 **TXABT:** Message Aborted Status bit<sup>(2,3)</sup>

1 = Message was aborted

0 = Message completed successfully

bit 6 **TXLARB:** Message Lost Arbitration Status bit<sup>(2,3)</sup>

1 = Message lost arbitration while being sent

0 = Message did not loose arbitration while being sent

bit 5 **TXERR:** Error Detected During Transmission bit<sup>(2,3)</sup>

1 = A bus error occurred while the message was being sent

0 = A bus error did not occur while the message was being sent

bit 4 **TXATIF:** Transmit Attempts Exhausted Interrupt Pending bit

TXEN = 1: (FIFO configured as a Transmit Buffer)

1 = Interrupt pending

0 = Interrupt Not pending

TXEN = 0: (FIFO configured as a Receive Buffer)

Unused, read as '0'

bit 3 **RXOVIF:** Receive FIFO Overflow Interrupt Flag bit

TXEN = 1: (FIFO configured as a Transmit Buffer)

Unused, read as '0'

TXEN = 0: (FIFO configured as a Receive Buffer)

1 = Overflow event has occurred

0 = No overflow event occurred

**Note 1:** FIFOCI<4:0> gives a zero-indexed value to the message in the FIFO. If the FIFO is four messages deep (FSIZE = 5'h03), FIFOCI will take on a value of 0 to 3 depending on the state of the FIFO.

**2:** This bit is reset on any read of this register or when the Transmit Queue is reset.

**3:** This bit is updated when a message completes (or aborts) or when the FIFO is reset.

**4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-18: CFD1TXQSTAn: TRANSMIT QUEUE STATUS REGISTER ('n' = 1-31) (4) (CONTINUED)

- bit 2     **TFERFFIF**: Transmit/Receive FIFO Empty/Full Interrupt Flag bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Empty Interrupt Flag  
          1 = FIFO is empty  
          0 = FIFO is not empty, at least 1 message queued to be transmitted  
  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Full Interrupt Flag  
          1 = FIFO is full  
          0 = FIFO is not full
- bit 1     **TFHRFHIF**: Transmit/Receive FIFO Half Empty/Half Full Interrupt Flag bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Half Empty Interrupt Flag  
          1 = FIFO is less than or equal to half full  
          0 = FIFO is greater than half full  
  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Half Full Interrupt Flag  
          1 = FIFO is greater than or equal to half full  
          0 = FIFO is less than half full
- bit 0     **TFNRFNIF**: Transmit/Receive FIFO Not Full/Not Empty Interrupt Flag bit  
          TXEN = 1: (FIFO configured as a Transmit FIFO)  
          Transmit FIFO Not Full Interrupt Flag  
          1 = FIFO is not full  
          0 = FIFO is full  
  
          TXEN = 0: (FIFO configured as a Receive FIFO)  
          Receive FIFO Not Empty Interrupt Flag  
          1 = FIFO is not empty (has at least one message)  
          0 = FIFO is empty

- Note 1:** FIFOCI<4:0> gives a zero-indexed value to the message in the FIFO. If the FIFO is four messages deep (FSIZE = 5'h03), FIFOCI will take on a value of 0 to 3 depending on the state of the FIFO.
- 2:** This bit is reset on any read of this register or when the Transmit Queue is reset.
- 3:** This bit is updated when a message completes (or aborts) or when the FIFO is reset.
- 4:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-19: CFD1TEFCON: TRANSMIT EVENT FIFO CONTROL REGISTER(2)**

| Bit Range | Bit 31/23/15/7            | Bit 30/22/14/6 | Bit 29/21/13/5                  | Bit 28/20/12/4 | Bit 27/19/11/3   | Bit 26/18/10/2    | Bit 25/17/9/1   | Bit 24/16/8/0    |
|-----------|---------------------------|----------------|---------------------------------|----------------|------------------|-------------------|-----------------|------------------|
| 31:24     | U-0<br>—                  | U-0<br>—       | U-0<br>—                        | R/W-0          | R/W-0            | R/W-0             | R/W-0           | R/W-0            |
|           | FSIZE<4:0> <sup>(1)</sup> |                |                                 |                |                  |                   |                 |                  |
| 23:16     | U-0<br>—                  | U-0<br>—       | U-0<br>—                        | U-0<br>—       | U-0<br>—         | U-0<br>—          | U-0<br>—        | U-0<br>—         |
| 15:8      | U-0<br>—                  | U-0<br>—       | U-0<br>—                        | U-0<br>—       | U-0<br>—         | S, HC-1<br>FRESET | U-0<br>—        | S, HC-0<br>UINC  |
| 7:0       | U-0<br>—                  | U-0<br>—       | R/W-0<br>TEFTSEN <sup>(1)</sup> | U-0<br>—       | R/W-0<br>TEFOVIE | R/W-0<br>TEFFIE   | R/W-0<br>TEFHIE | R/W-0<br>TEFNEIE |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-29 **Unimplemented:** Read as '0'

bit 28-24 **FSIZE<4:0>:** FIFO Size bits<sup>(1)</sup>

- 11111 = FIFO is 32 Messages deep
- .
- .
- .
- 00010 = FIFO is 3 Messages deep
- 00001 = FIFO is 2 Messages deep
- 00000 = FIFO is 1 Message deep

bit 23-11 **Unimplemented:** Read as '0'

bit 10 **FRESET:** FIFO Reset bit

- 1 = FIFO will be reset when bit is set, cleared by hardware when FIFO is reset. The user should poll this bit is clear before taking any action
- 0 = No effect

bit 9 **Unimplemented:** Read as '0'

bit 8 **UINC:** Increment Tail bit

When this bit is set the FIFO tail will increment by a single message

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **TEFTSEN:** Transmit Event FIFO Time Stamp Enable bit<sup>(1)</sup>

- 1 = Time stamp elements in TEF
- 0 = Do not time stamp elements in TEF

bit 4 **Unimplemented:** Read as '0'

bit 3 **TEFOVIE:** Transmit Event FIFO Overflow Interrupt Enable bit

- 1 = Interrupt enabled for overflow event
- 0 = Interrupt disabled for overflow event

bit 2 **TEFFIE:** Transmit Event FIFO Full Interrupt Enable bit

- 1 = Interrupt enabled for FIFO full
- 0 = Interrupt disabled for FIFO full

bit 1 **TEFHIE:** Transmit Event FIFO Half Full Interrupt Enable bit

- 1 = Interrupt enabled for FIFO half full
- 0 = Interrupt disabled for FIFO half full

**Note 1:** These bits can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).

**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-19: CFD1TEFCON: TRANSMIT EVENT FIFO CONTROL REGISTER(2)

bit 0      **TEFNEIE:** Transmit Event FIFO Not Empty Interrupt Enable bit  
1 = Interrupt enabled for FIFO not empty  
0 = Interrupt disabled for FIFO not empty

- Note 1:** These bits can only be modified in Configuration mode (OPMOD<2:0> bits (CFDxCON<23:21>) = 100).  
**2:** Not available on PIC32MK GPG variants.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-20: CFD1TEFSTA: TRANSMIT EVENT FIFO STATUS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2        | Bit 25/17/9/1         | Bit 24/16/8/0          |
|-----------|----------------|----------------|----------------|----------------|----------------|-----------------------|-----------------------|------------------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0                   | U-0                   | U-0                    |
|           | —              | —              | —              | —              | —              | —                     | —                     | —                      |
| 23:16     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0                   | U-0                   | U-0                    |
|           | —              | —              | —              | —              | —              | —                     | —                     | —                      |
| 15:8      | U-0            | U-0            | U-0            | U-0            | U-0            | U-0                   | U-0                   | U-0                    |
|           | —              | —              | —              | —              | —              | —                     | —                     | —                      |
| 7:0       | U-0            | U-0            | U-0            | U-0            | HS, C-0        | R-0                   | R-0                   | R-0                    |
|           | —              | —              | —              | —              | TEFOVIF        | TEFFIF <sup>(1)</sup> | TEFHIF <sup>(1)</sup> | TEFNEIF <sup>(1)</sup> |

|                   |                   |                                    |
|-------------------|-------------------|------------------------------------|
| <b>Legend:</b>    | C = Clearable bit | HS = Set by hardware               |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared               |
|                   |                   | x = Bit is unknown                 |

bit 31-4 **Unimplemented:** Read as '0'

bit 3 **TEFOVIF:** Transmit Event FIFO Overflow Interrupt Flag bit

- 1 = Overflow event has occurred
- 0 = No overflow event has occurred

bit 2 **TEFFIF:** Transmit Event FIFO Full Interrupt Flag bit<sup>(1)</sup>

- 1 = FIFO is full
- 0 = FIFO is not full

bit 1 **TEFHIF:** Transmit Event FIFO Half Full Interrupt Flag bit<sup>(1)</sup>

- 1 = FIFO is greater than or equal to half full
- 0 = FIFO is less than half full

bit 0 **TEFNEIF:** Transmit Event FIFO Not Empty Interrupt Flag bit<sup>(1)</sup>

- 1 = FIFO is not empty (has at least one message)
- 0 = FIFO is empty

**Note 1:** This bit is read-only and reflects the status of the FIFO.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-21: CFD1FIFOAn: DEFINITION REGISTER ('n' = 1-31)

| Bit Range    | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|--------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24        | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| FIFOA<31:24> |                |                |                |                |                |                |               |               |
| 23:16        | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| FIFOA<23:16> |                |                |                |                |                |                |               |               |
| 15:8         | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| FIFOA<15:8>  |                |                |                |                |                |                |               |               |
| 7:0          | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| FIFOA<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **FIFOA<31:0>**: FIFO User Address bits

TXEN = 1: (FIFO configured as a Transmit Buffer)

A read of this register will return the address where the next message is to be written (FIFO head).

TXEN = 0: (FIFO configured as a Receive Buffer)

A read of this register will return the address where the next message is to be read (FIFO tail).

**Note 1:** This register is not guaranteed to read correctly in Configuration Mode and should only be accessed when the module is *not* in Configuration Mode.

**2:** This register provides the byte address in the message memory of the next element in the FIFO. The application uses this address directly to access RAM.

- For a RX FIFO, the address points to the next element the application should read from

- For a TX FIFO, the address points to the next element, the application should write to

After accessing this register, the user application must set the UINC bit in the CFDxFIFOCONn register, which will update the FIFO pointer.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-22: CFD1TEFUA: TRANSMIT EVENT FIFO USER ADDRESS REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | TEFUA<31:24>   |                |                |                |                |                |               |               |
| 23:16     | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | TEFUA<23:16>   |                |                |                |                |                |               |               |
| 15:8      | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | TEFUA<15:8>    |                |                |                |                |                |               |               |
| 7:0       | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
|           | TEFUA<7:0>     |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **TEFUA<31:0>**: Transmit Event FIFO User Address bits

A read of this register will return the address where the next event is to be read (FIFO tail).

**Note 1:** This register is not guaranteed to read correctly in Configuration Mode and should only be accessed when the module is *not* in Configuration Mode.

- 2:** Elements in the Transmit Event FIFO can be accessed through this register. The register provides the byte address in the message memory of the next element in the buffer. The application uses this address directly to access RAM. The address points to the next element the application should read from. After accessing this register, the user application must set the UINC bit in the CFDxTEFCON register, which will update the FIFO pointer.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-23: CFD1TXQUA: TRANSMIT QUEUE USER ADDRESS REGISTER

| Bit Range    | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|--------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24        | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| TXQUA<31:24> |                |                |                |                |                |                |               |               |
| 23:16        | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| TXQUA<23:16> |                |                |                |                |                |                |               |               |
| 15:8         | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| TXQUA<15:8>  |                |                |                |                |                |                |               |               |
| 7:0          | R-x            | R-x            | R-x            | R-x            | R-x            | R-x            | R-x           | R-x           |
| TXQUA<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **TXQUA<31:0>**: Transmit Queue User Address bits

A read of this register will return the address where the next message is to be written (Transmit Queue head).

**Note 1:** This register is not guaranteed to read correctly in Configuration Mode and should only be accessed when the module is *not* in Configuration Mode.

- 2:** Elements in the Transmit Queue can be accessed through this register. The register provides the byte address in the message memory of the next element in the Transmit Queue. The application uses this address directly to access RAM. The address points to the next element, the application should write to. After accessing this register, the user application must set the UINC bit in the CFDxTXQCON register, which will update the TXQ pointer.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-24: CFD1TREC: TRANSMIT/RECEIVE ERROR COUNT REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | R-1            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | —              | —              | TXBO           | TXBP           | RXBP           | TXWARN         | RXWARN        | EWARN         |
| 15:8      | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | TERRCNT<7:0>   |                |                |                |                |                |               |               |
| 7:0       | R-0            | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | RERRCNT<7:0>   |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-22 **Unimplemented:** Read as '0'

bit 21 **TXBO:** Transmitter in Error State Bus Off bit (TERRCNT > 255)  
In Configuration mode, TXBO is set, since the CAN FD module is not on the bus.  
1 = Indicates that the number of transmit errors is greater than 255. As a result, the CAN FD module will automatically enter Bus Off mode.  
0 = Indicates that transmit errors are less than 255

bit 20 **TXBP:** Transmitter in Error State Bus Passive bit (TERRCNT > 127)  
1 = Indicates that the number of transmit errors is greater than 127. As a result, the CAN FD module will automatically enter Bus Passive mode.  
0 = Indicates that transmit errors are less than or equal to 127

bit 19 **RXBP:** Receiver in Error State Bus Passive bit (RERRCNT > 127)  
1 = Indicates that the number of receive errors is greater than 127. As a result, the CAN FD module will automatically enter Bus Passive mode.  
0 = Indicates that transmit errors are less than or equal to 127

bit 18 **TXWARN:** Transmitter in Error State Warning bit (128 > TERRCNT > 95)  
1 = Indicates that the number of transmit errors are less than 128, but are greater than 95.  
0 = Indicates that transmit errors are greater than or equal to 95

bit 17 **RXWARN:** Receiver in Error State Warning bit (128 > RERRCNT > 95)  
1 = Indicates that the number of receives errors is less than 128, but are greater than 95.  
0 = Indicates that receive errors are greater than or equal to 95

bit 16 **EWARN:** Transmitter or Receiver is in Error State Warning bit  
1 = TXWARN or RXWARN is set  
0 = No transmit or receive warnings

**Note:** Separate error counters for arbitration and data phase; receive and transmit:  
• Successful message counter  
• Keep track of received ESI, (Error State Indicator)

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-24: CFD1TREC: TRANSMIT/RECEIVE ERROR COUNT REGISTER

bit 15-8 **TERRCNT<7:0>**: Transmit Error Counter bits

11111111 = Greater than or equal to 255 Transmit errors. The TXBO bit is equal to '1' if the TERRCNT<7:0> bits are greater than 255.

11111110 = 254 Transmit errors

•  
•  
•

00000001 = 1 transmit error

00000000 = No transmit errors

**Note:** The following conditions apply:

- If  $128 > \text{TERRCNT} > 95$ , the TXWARN bit = 1
- If  $\text{TERRCNT} > 95$ , the TXBP bit = 1 and TX enters Bus Passive mode
- If  $\text{TERRCNT} > 255$ , the TXBO and CERRIF bits are set and the CAN FD module enters Bus Off mode and starts the Bus Off recovery sequence

bit 7-0 **RERRCNT<7:0>**: Receive Error Counter bits

11111111 = Greater than or equal to 255 Receive errors

11111110 = 254 receive errors

•  
•  
•

00000001 = 1 receive error

00000000 = No receive errors

**Note:** The following conditions apply:

- If  $\text{RERRCNT} > 127$ , the RXBP bit = 1
- If  $128 > \text{RERRCNT} > 95$ , the RXWARN bit = 1

**Note:** Separate error counters for arbitration and data phase; receive and transmit:

- Successful message counter
- Keep track of received ESI, (Error State Indicator)

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-25: CFD1BDIAG0: BUS DIAGNOSTICS REGISTER 0

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DTERRCNT<7:0>  |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | DRERRCNT<7:0>  |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NTERRCNT<7:0>  |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | NRERRCNT<7:0>  |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-24 **DTERRCNT<7:0>**: Data Bit Rate Transmit Error Counter bits

bit 23-16 **DRERRCNT<7:0>**: Data Bit Rate Receive Error Counter bits

bit 15-8 **NTERRCNT<7:0>**: Nominal Bit Rate Transmit Error Counter bits

bit 7-0 **NRERRCNT<7:0>**: Nominal Bit Rate Receive Error Counter bits

**Note:** This register keeps track of bus errors during nominal and data bit rate phases, separately. These counters work differently than in the CFDxTREC register:

- Counters are incremented (+1) on any bus error
- Counters are not decremented
- Counters are cleared on a register read

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-26: CFD1BDIAG1: BUS DIAGNOSTICS REGISTER 1

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | U-0            | R/W-0         | R/W-0         |
|           | DLCMM          | ESI            | DCRCERR        | DSTUFERR       | DFORMERR       | —              | DBIT1ERR      | DBIT0ERR      |
| 23:16     | R/W-0          | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | TXBOERR        | —              | NCRCERR        | NSTUFERR       | NFORMERR       | NACKERR        | NBIT1ERR      | NBIT0ERR      |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EFMSGCNT<15:8> |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EFMSGCNT<7:0>  |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

- bit 31     **DLCMM:** Data Length Code (DLC) Mismatch Status bit  
 1 = Payload size error. During a transmission or reception the Data Length Code exceeds the payload size of the FIFO element defined in the PLSIZE<2:0> bits (CFDxFIFOCONn<31:29>).  
 0 = No payload size error occurred
- bit 30     **ESI:** Error State Indicator (ESI) Flag Status bit  
 1 = ESI flag of a received CAN FD message was set  
 0 = ESI flag of a received CAN FD message was not set
- bit 29     **DCRCERR:** Data CRC Error Status bit  
 1 = CRC checksum of a received message was incorrect. The CRC of an incoming message does not match with the CRC calculated from the received data  
 0 = No received message data CRC error occurred
- bit 28     **DSTUFERR:** Data Stuffing Error Status bit  
 1 = More than five equal bits in a sequence have occurred in a portion of a received message where this is not allowed  
 0 = No data received message errors
- bit 27     **DFORMERR:** Data Format Error Status bit  
 1 = Data fixed format portion of a received frame has the wrong format  
 0 = No format errors occurred
- bit 26     **Unimplemented:** Read as '0'
- bit 25     **DBIT1ERR:** Data Bit Logical '1' Error Status bit  
 1 = During the data transmission of a message (with the exception of the arbitration field), the device wanted to send a recessive level (bit of logical value '1'), but the monitored bus value was dominant.  
 0 = No data logical '1' bit transmission error occurred

**Note:** This register shows the type of errors that occurred since the last read. Corresponding bits are set when an error occurs, but *all* bits are cleared on any read or R-M-W bit manipulation instruction. Errors are separately tracked for data and nominal bit rate phases. The Error Free Message Counter bits (EFMSGCNT<15:0>), together with the Error Counters and the Error Flags can be used to determine the quality of the bus.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-26: CFD1BDIAG1: BUS DIAGNOSTICS REGISTER 1 (CONTINUED)

- bit 24 **DBIT0ERR:** Data Bit Logical '0' Error Status bit  
1 = During the data transmission of a message (or acknowledge bit, or active error flag, or overload flag), the device wanted to send a dominant level (data or identifier bit logical value '0'), but the monitored bus value was recessive. During Bus Off recovery, this status is set each time a sequence of 11 recessive bits has been monitored. This enables the CPU to monitor the proceeding of the Bus Off recovery sequence (indicating the bus is not stuck at dominant or continuously disturbed).  
0 = No data logical '0' bit transmission error occurred
- bit 23 **TXBOERR:** Transmit Bus Off Error Status bit  
1 = Transmit error occurred and device enter Bus Off mode and automatically recovered  
0 = No transmit Bus Off error occurred
- bit 22 **Unimplemented:** Read as '0'
- bit 21 **NCRCERR:** Nominal CRC Error Status bit  
1 = The CRC checksum of a nominal received message was incorrect. The CRC of an incoming nominal message does not match with the CRC calculated from the received data.  
0 = No nominal CRC received message error occurred
- bit 20 **NSTUFERR:** Nominal Stuffing Error Status bit  
1 = More than five equal bits in a sequence have occurred in a part of a nominal received message where this is not allowed  
0 = No nominal bit stuffing errors have occurred
- bit 19 **NFORMERR:** Nominal Format Error Status bit  
1 = A fixed format portion of a nominal received frame has the wrong format  
0 = No nominal format error occurred
- bit 18 **NACKERR:** Not Acknowledged Error Status bit  
1 = A transmitted message was not acknowledged  
0 = A transmitted message was acknowledged
- bit 17 **NBIT1ERR:** Nominal Bit Logical '1' Error Status bit  
1 = During the transmission of a nominal message (with the exception of the arbitration field), the device wanted to send a recessive level (bit of logical value '1'), but the monitored bus value was dominant  
0 = No nominal bit logical '1' error occurred
- bit 16 **NBIT0ERR:** Nominal Bit Logical '0' Error Status bit  
1 = During the transmission of a nominal message (or acknowledge bit, or active error flag, or overload flag), the device wanted to send a dominant level (data or identifier bit logical value '0'), but the monitored bus value was recessive. During Bus Off recovery, this status is set each time a sequence of 11 recessive bits has been monitored. This enables the CPU to monitor the proceeding of the Bus Off recovery sequence (indicating the bus is not stuck at dominant or continuously disturbed).  
0 = No nominal bit logical '0' error occurred
- bit 15-0 **EFMSGCNT<15:0>:** Error Free Message Counter Status bits  
1111111111111111 = 65,536 error free messages since the last register read  
.  
.  
.  
0000000000000000 = 0 error free messages since the last register read

**Note:** The EFMSGCNT<15:0> bits increment on any error free message on the bus. These bits are cleared on any read of this register.

**Note:** This register shows the type of errors that occurred since the last read. Corresponding bits are set when an error occurs, but *all* bits are cleared on any read or R-M-W bit manipulation instruction. Errors are separately tracked for data and nominal bit rate phases. The Error Free Message Counter bits (EFMSGCNT<15:0>), together with the Error Counters and the Error Flags can be used to determine the quality of the bus.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-27: CFD1FLTCON0: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4           | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|--------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN3         | —              | —              | F3BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN2         | —              | —              | F2BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN1         | —              | —              | F1BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN0         | —              | —              | F0BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN3:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented:** Read as '0'

bit 28-24 **F3BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN2:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented:** Read as '0'

bit 20-16 **F2P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN1** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-27: CFD1FLTCON0: FILTER CONTROL REGISTER (CONTINUED)

bit 12-8 **F1BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 7 **FLTEN0**: Enable Filter n to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 6-5 **Unimplemented**: Read as '0'

bit 4-0 **F0BP<4:0>**: Pointer to Object when Filter 'n' hits bits <sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-28: CFD1FLTCON1: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4           | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|--------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN7         | —              | —              | F7BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN6         | —              | —              | F6BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN5         | —              | —              | F5BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                    | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN4         | —              | —              | F4BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN7**: Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented**: Read as '0'

bit 28-24 **F7BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN6**: Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented**: Read as '0'

bit 20-16 **F6P<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN5** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented**: Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-28: CFD1FLTCON1: FILTER CONTROL REGISTER (CONTINUED)

bit 12-8 **F5BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 7 **FLTEN4**: Enable Filter n to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 6-5 **Unimplemented**: Read as '0'

bit 4-0 **F4BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-29: CFD1FLTCON2: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN11        | —              | —              | F11BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN10        | —              | —              | F10BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN9         | —              | —              | F9BP<4:0> <sup>(1)</sup>  |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN8         | —              | —              | F8BP<4:0> <sup>(1)</sup>  |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN11:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented:** Read as '0'

bit 28-24 **F11BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN10:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented:** Read as '0'

bit 20-16 **F10P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN9** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-29: CFD1FLTCON2: FILTER CONTROL REGISTER (CONTINUED)

bit 12-8 **F9BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 7 **FLTEN8**: Enable Filter n to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 6-5 **Unimplemented**: Read as '0'

bit 4-0 **F8BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-30: CFD1FLTCON3: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN15        | —              | —              | F15BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN4         | —              | —              | F14BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN13        | —              | —              | F13BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN12        | —              | —              | F12BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31     **FLTEN15:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 30-29     **Unimplemented:** Read as '0'

bit 28-24     **F15BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31  
 11110 = Message matching filter is stored in Object 30  
 .  
 .  
 .  
 00010 = Message matching filter is stored in Object 2  
 00001 = Message matching filter is stored in Object 1  
 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23     **FLTEN14:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 22-21     **Unimplemented:** Read as '0'

bit 20-16     **F14P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31  
 11110 = Message matching filter is stored in Object 30  
 .  
 .  
 .  
 00010 = Message matching filter is stored in Object 2  
 00001 = Message matching filter is stored in Object 1  
 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15     **FLTEN13** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 14-13     **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-30: CFD1FLTCON3: FILTER CONTROL REGISTER (CONTINUED)

- bit 12-8 **F13BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.
- bit 7 **FLTEN12**: Enable Filter n to Accept Messages bits
- 1 = Filter is enabled
  - 0 = Filter is disabled
- bit 6-5 **Unimplemented**: Read as '0'
- bit 4-0 **F12BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-31: CFD1FLTCON4: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN19        | —              | —              | F19BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN18        | —              | —              | F18BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN17        | —              | —              | F17BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN16        | —              | —              | F16BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN19:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented:** Read as '0'

bit 28-24 **F19BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN18:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented:** Read as '0'

bit 20-16 **F18P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN17:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-31: CFD1FLTCON4: FILTER CONTROL REGISTER (CONTINUED)

- bit 12-8 **F17BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.
- bit 7 **FLTEN16**: Enable Filter n to Accept Messages bits
- 1 = Filter is enabled
  - 0 = Filter is disabled
- bit 6-5 **Unimplemented**: Read as '0'
- bit 4-0 **F16BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-32: CFD1FLTCON5: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN23        | —              | —              | F23BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN22        | —              | —              | F22BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN21        | —              | —              | F21BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN20        | —              | —              | F20BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31     **FLTEN23:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 30-29     **Unimplemented:** Read as '0'

bit 28-24     **F23BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31  
 11110 = Message matching filter is stored in Object 30  
 .  
 .  
 .  
 00010 = Message matching filter is stored in Object 2  
 00001 = Message matching filter is stored in Object 1  
 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23     **FLTEN22:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 22-21     **Unimplemented:** Read as '0'

bit 20-16     **F22P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31  
 11110 = Message matching filter is stored in Object 30  
 .  
 .  
 .  
 00010 = Message matching filter is stored in Object 2  
 00001 = Message matching filter is stored in Object 1  
 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15     **FLTEN21** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled  
 0 = Filter is disabled

bit 14-13     **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-32: CFD1FLTCON5: FILTER CONTROL REGISTER (CONTINUED)

- bit 12-8 **F21BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.
- bit 7 **FLTEN20**: Enable Filter n to Accept Messages bits
- 1 = Filter is enabled
  - 0 = Filter is disabled
- bit 6-5 **Unimplemented**: Read as '0'
- bit 4-0 **F20BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>
- 11111 = Message matching filter is stored in Object 31
  - 11110 = Message matching filter is stored in Object 30
  - .
  - .
  - 00010 = Message matching filter is stored in Object 2
  - 00001 = Message matching filter is stored in Object 1
  - 00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-33: CFD1FLTCON6: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN27        | —              | —              | F27BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN26        | —              | —              | F26BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN25        | —              | —              | F25BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN24        | —              | —              | F24BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN27:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented:** Read as '0'

bit 28-24 **F27P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN26:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented:** Read as '0'

bit 20-16 **F25P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN25** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-33: CFD1FLTCON6: FILTER CONTROL REGISTER (CONTINUED)

bit 12-8 **F25BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 7 **FLTEN24**: Enable Filter n to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 6-5 **Unimplemented**: Read as '0'

bit 4-0 **F24BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

•  
•  
•

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-34: CFD1FLTCON7: FILTER CONTROL REGISTER

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4            | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN31        | —              | —              | F31BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 23:16     | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN30        | —              | —              | F30BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 15:8      | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN29        | —              | —              | F29BP<4:0> <sup>(1)</sup> |                |                |               |               |
| 7:0       | R/W-0          | U-0            | U-0            | R/W-0                     | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | FLTEN28        | —              | —              | F28BP<4:0> <sup>(1)</sup> |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **FLTEN31:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 30-29 **Unimplemented:** Read as '0'

bit 28-24 **F31BP<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 23 **FLTEN30:** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 22-21 **Unimplemented:** Read as '0'

bit 20-16 **F30P<4:0>:** Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

.

.

.

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 15 **FLTEN29** Enable Filter 'n' to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 14-13 **Unimplemented:** Read as '0'

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 24-34: CFD1FLTCON7: FILTER CONTROL REGISTER (CONTINUED)

bit 12-8 **F29BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

·  
·  
·

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

bit 7 **FLTEN28**: Enable Filter n to Accept Messages bits

1 = Filter is enabled

0 = Filter is disabled

bit 6-5 **Unimplemented**: Read as '0'

bit 4-0 **F28BP<4:0>**: Pointer to Object when Filter 'n' hits bits<sup>(1)</sup>

11111 = Message matching filter is stored in Object 31

11110 = Message matching filter is stored in Object 30

·  
·  
·

00010 = Message matching filter is stored in Object 2

00001 = Message matching filter is stored in Object 1

00000 = Reserved. Object 0 is the TX Queue and cannot receive messages.

**Note 1:** These bits can only be modified if the corresponding filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 24-35: CFD1FLTOBJn: FILTER OBJECT REGISTER ('n' = 0-31)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | EXIDE          | SID11          | EID<17:13>     |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EID<12:5>      |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | EID<4:0>       |                |                |                |                | SID<10:8>      |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | SID<7:0>       |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31 **Unimplemented:** Read as '0'

bit 30 **EXIDE:** Extended Identifier Enable bit

If MIDE = 1:

1 = Match only messages with extended identifier addresses

0 = Match only messages with standard identifier addresses

bit 29 **SID11:** Standard Identifier Filter bit

1 = Extended standard identifier to 12-bit (i.e., SID<11:0>)

0 = Do not use extended standard Identifier bits

**Note:** Standard identifier filter bit RRS in the CAN FD base frame can be used to extend the SID to 12-bit. When enabled, it is referred to as SID11, which is the MSB of SID<11:0>.

bit 28-11 **EID<17:0>:** Extended Identifier Filter bits

In DeviceNet™ mode, these are the filter bits used in conjunction with the Device Net Filter Bit Number bits, DNCNT<4:0> (CFDxCON<4:0>).

bit 10-0 **SID<10:0>:** Standard Identifier filter bits

These are the standard ID message filter bits.

**Note:** This register can only be changed when the filter is disabled (FLTENx bits (CFDxFLTCONn) = 0).



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 24-36: CFD1MASKn: MASK REGISTER ('n' = 0-31)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | —              | MIDE           | MSID11         | MEID<17:13>    |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | MEID<12:5>     |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | MEID<4:0>      |                |                |                |                | MSID<10:8>     |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | MSID<7:0>      |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31 **Unimplemented:** Read as '0'

bit 30 **MIDE:** Identifier Receive Mode bit

1 = Match only message types (standard or extended address) that correspond to EXIDE bit in filter

0 = Match either standard or extended address message if filters match (i.e., if (Filter SID) = (Message SID) or if (Filter SID/EID) = (Message SID/EID))

bit 29 **MSID11:** Standard Identifier Mask bit

1 = Enable extended standard identifier mask to 12 bits (MSID<10:0>, MEID<0>)

0 = Do not enable extended standard identifier mask (MSID<10:0>)

bit 28-11 **MEID<17:0>:** Extended Identifier Mask bits

In DeviceNet™ mode, these are the mask bits for the extended CAN ID, which are the first two data bytes.

bit 10-0 **MSID<10:0>:** Standard Identifier Mask bits

These are the standard CAN message identifier mask bits.

**Note:** This register can only be changed when the filter is disabled (FLTENx bits (CFDxFLTCOn = 0)).

# PIC32MK GPG/MCJ with CAN FD Family

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 25.0 OP AMP/COMPARATOR MODULE

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 39. “Op amp/Comparator”** (DS60001178), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

Depending on the device, the op amp/comparator module consists of a Comparator and op amp modules. When available, the op amps can be independently enabled or disabled from the Comparator.

The following are key features of the comparator module:

- Differential inputs
- Rail-to-rail operation
- Selectable output and trigger event polarity
- Selectable inputs:
  - Analog inputs multiplexed with I/O pins
  - On-chip internal voltage reference via a 12-bit CDAC output or an external pin
- Output debounce or Digital noise filter with these selectable clocks:
  - Peripheral Bus Clock (PBCLK2)
  - System Clock (SYSCLK)
  - Reference Clock 3 (REFCLK3)
  - PBCLK2/Timer PRx ('x' = 2-5)
  - PWM Secondary Special Event
- Outputs can be internally configured as trigger sources

The following are key features of the op amps:

- Inverting and non-inverting Inputs and output accessible on pins
- Rail-to-rail operation ( $3V \leq AV_{DD} \leq 3.6V$ )
- Internal connection to ADC Sample and Hold circuits/SAR cores
- Special voltage follower mode for buffering signals
- Low-power mode
- 10 ma IOL/IOH (Sink/Source)

Please refer to the PIC32MK GP Family Features in **TABLE 1: “PIC32MK Motor Control and General Purpose (MC and GP) Family Features”** for the actual number of available op amp/Comparator modules on your specific device.

Block diagrams of the op amp/comparator module are illustrated in [Figure 25-1](#) through [Figure 25-5](#).

**Note:** The Op amps are disabled by default (i.e., OPAXMD bit in the PMD2 register is equal to '1') on any Reset. Before use or access to any corresponding Op amp, ensure that the OPAXMD bit is equal to '0'.

**FIGURE 25-1: OP AMP 1/COMPARATOR 1 MODULE BLOCK DIAGRAM**

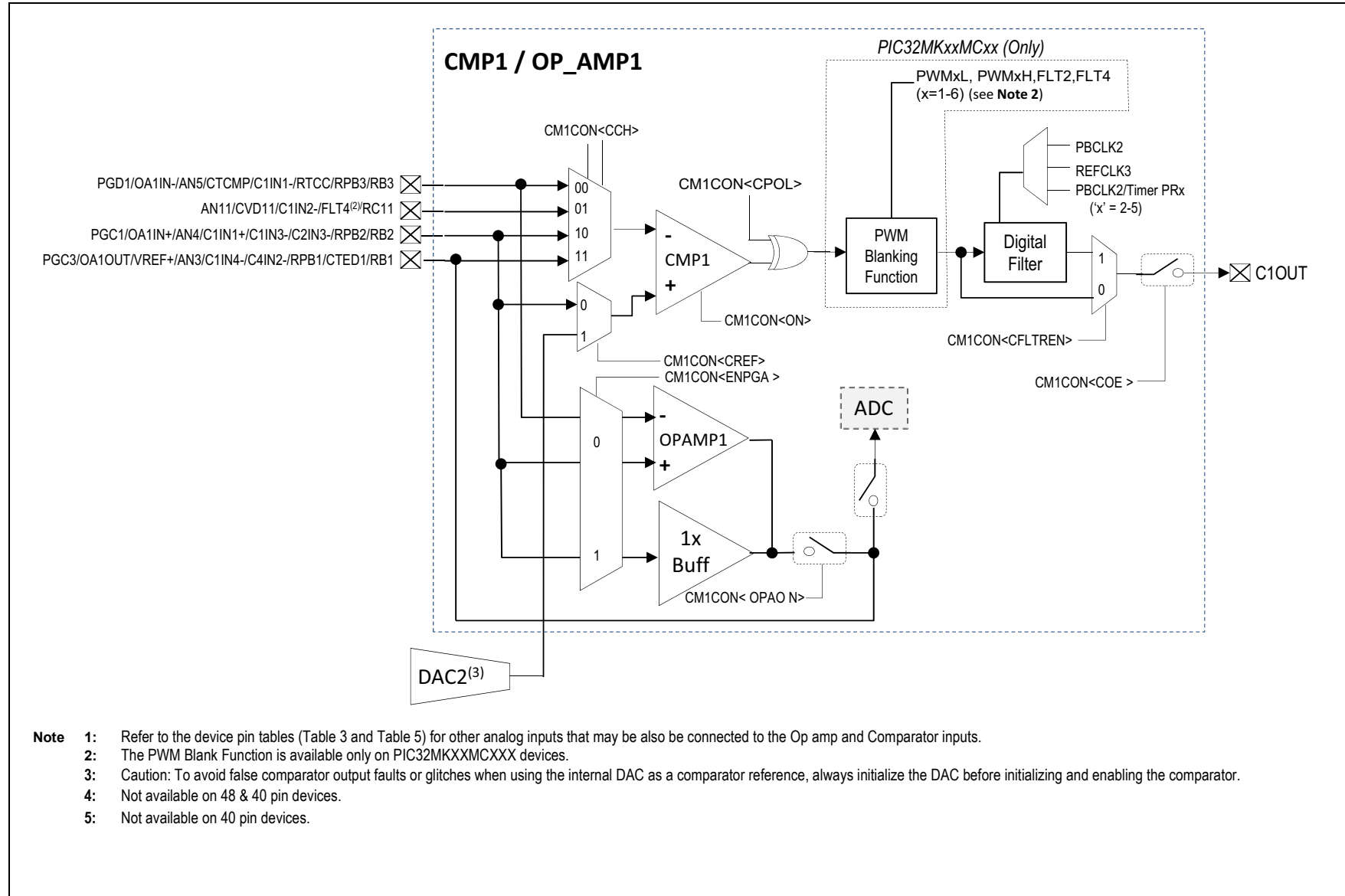
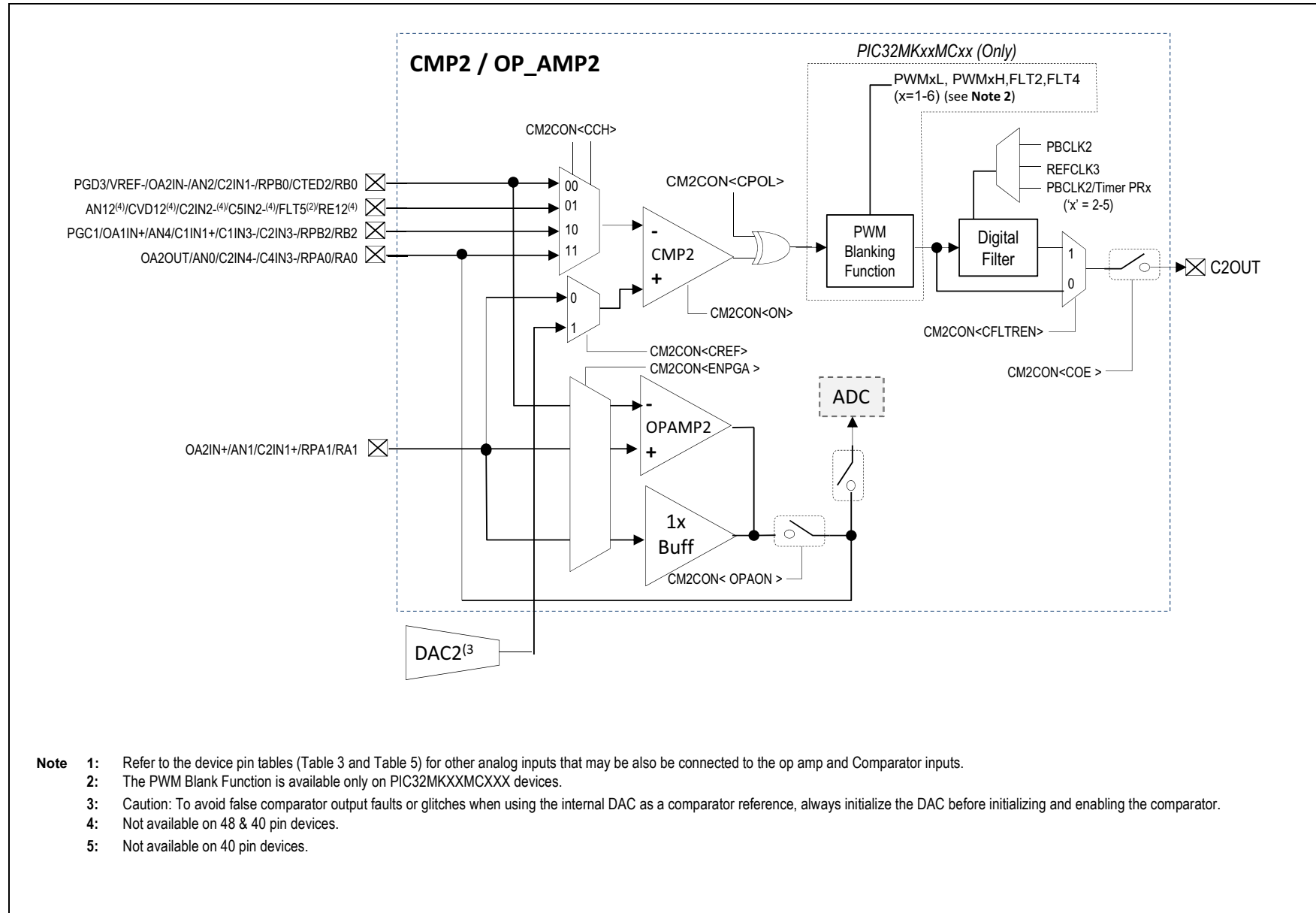
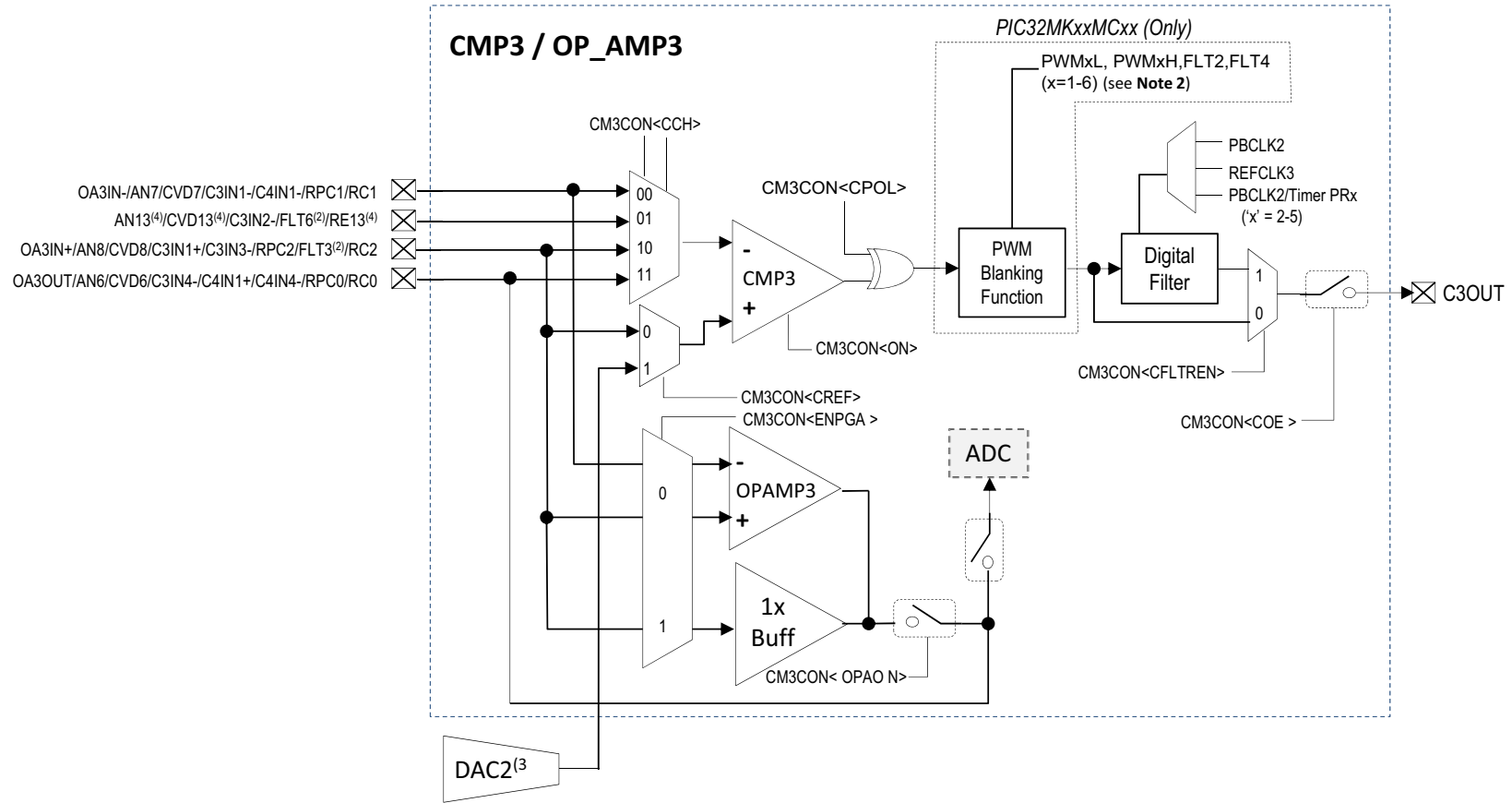


FIGURE 25-2: OP AMP 2/COMPARATOR 2 MODULE BLOCK DIAGRAM

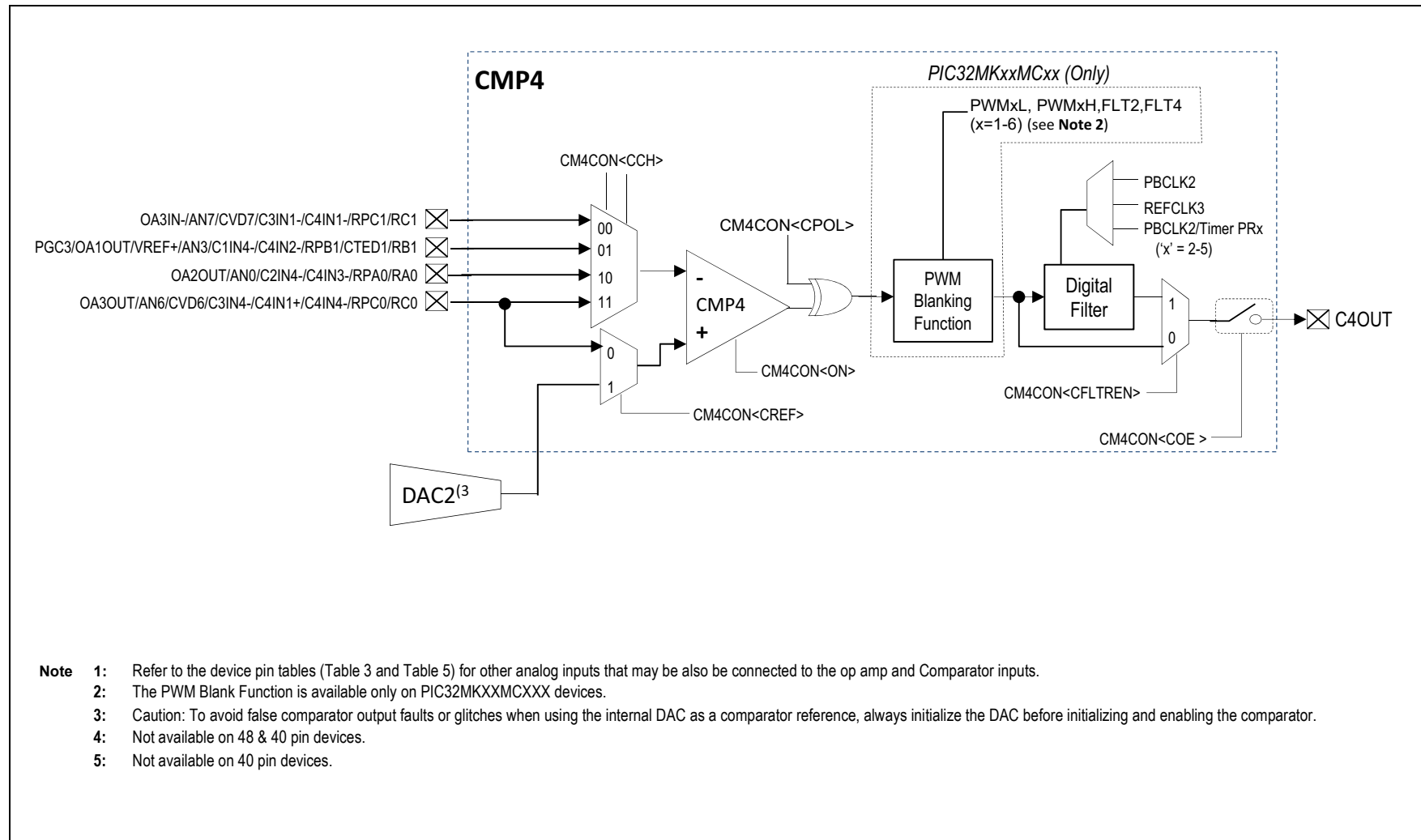


**FIGURE 25-3: OP AMP 3/COMPARATOR 3 MODULE BLOCK DIAGRAM**



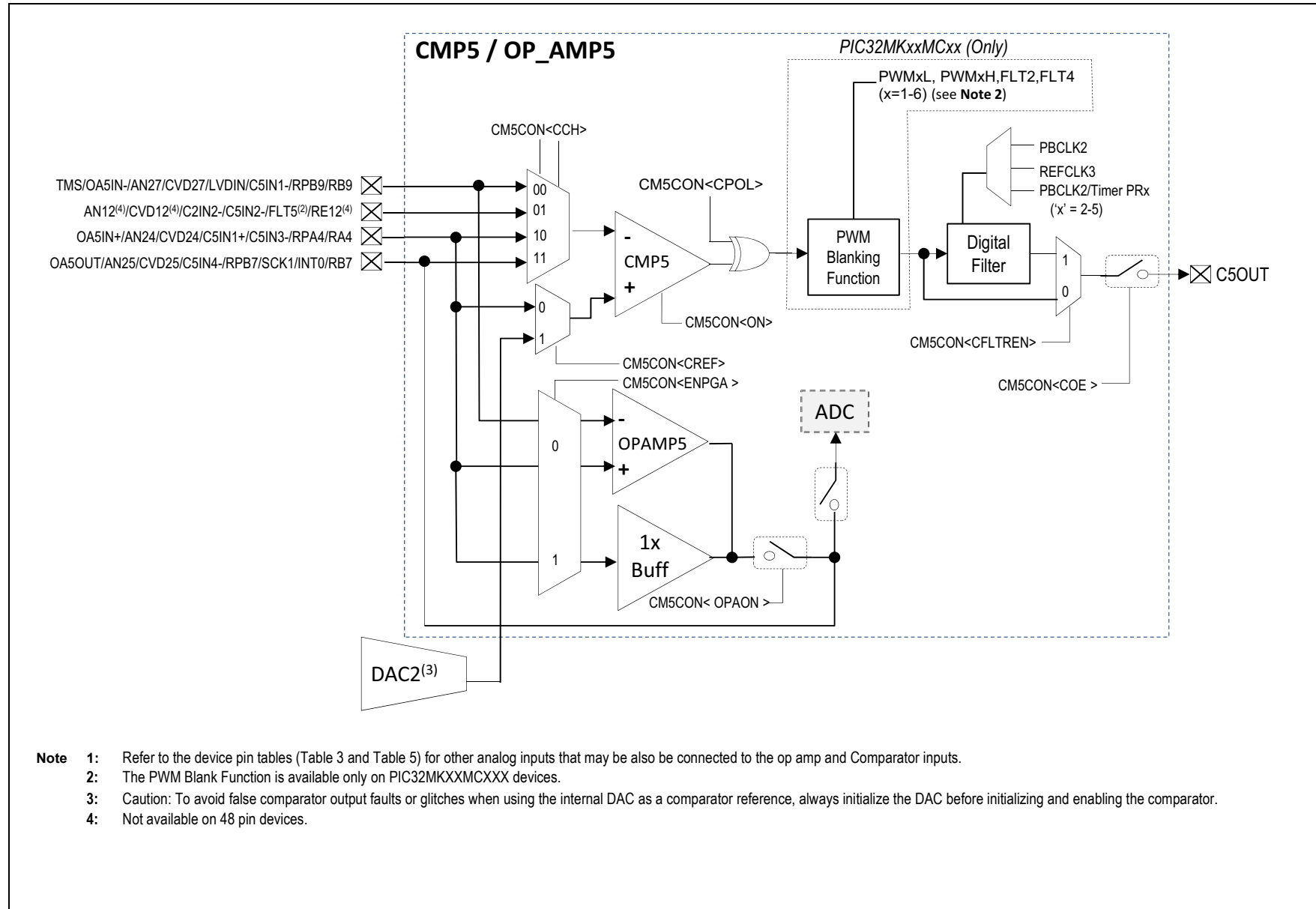
- Note**
- 1: Refer to the device pin tables (Table 3 and Table 5) for other analog inputs that may be also be connected to the op amp and Comparator inputs.
  - 2: The PWM Blank Function is available only on PIC32MKXXMCXXX devices.
  - 3: Caution: To avoid false comparator output faults or glitches when using the internal DAC as a comparator reference, always initialize the DAC before initializing and enabling the comparator.
  - 4: Not available on 48 & 40 pin devices.
  - 5: Not available on 40 pin devices.

**FIGURE 25-4: COMPARATOR 4 MODULE BLOCK DIAGRAM**



- Note**
- 1: Refer to the device pin tables (Table 3 and Table 5) for other analog inputs that may be also be connected to the op amp and Comparator inputs.
  - 2: The PWM Blank Function is available only on PIC32MKXXMCXXX devices.
  - 3: Caution: To avoid false comparator output faults or glitches when using the internal DAC as a comparator reference, always initialize the DAC before initializing and enabling the comparator.
  - 4: Not available on 48 & 40 pin devices.
  - 5: Not available on 40 pin devices.

**FIGURE 25-5: OP AMP 5/COMPARATOR 5 MODULE BLOCK DIAGRAM**





# PIC32MK GPG/MCJ with CAN FD Family

## 25.1 Comparator Interface

The Comparators also have both their inverting and non-inverting inputs accessible via device pins. The non-inverting input pins can be connected to an internal 12-bit CDAC to generate a precise reference or to an external reference through a pin. These references can be individually selected for each comparator module. The inverting inputs can be connected to one of four external pins or internally to outputs of the op amps. The Comparator outputs can be entirely disabled from appearing on the output pins, which relieves a pin for other uses, remapped to different pins via the peripheral pin select module, and selected to active-high or active-low polarity.

In Comparator modules that do not implement the op amp, the Comparator module has a different input selection configuration.

The stand-alone Comparator implements a 4 x 1 multiplexer at the inverting input to enable selection of the desired signal to compare against the non-inverting input. Up to three outputs of op amps can be internally connected to the Comparator through the multiplexer.

The Comparator may be enabled or disabled using the corresponding ON bit (CMxCON<15>) in the op amp/Comparator Control register. When the Comparator is disabled, the corresponding trigger and interrupt generation is disabled as well.

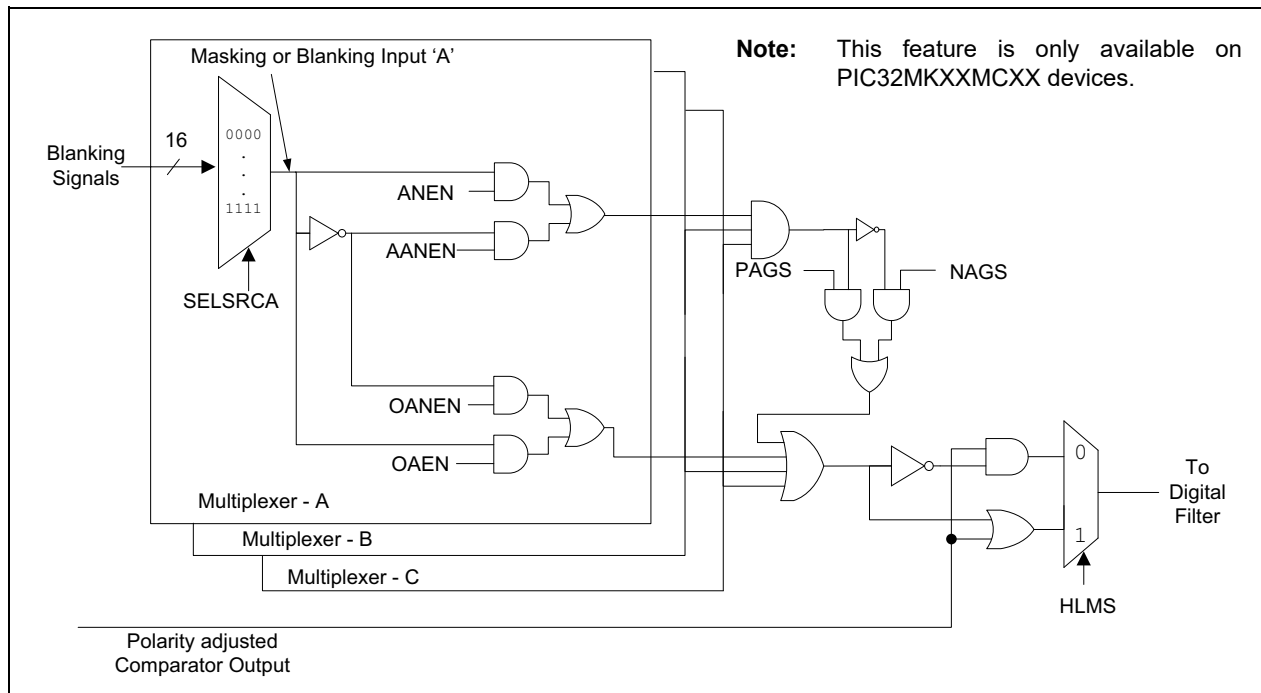
It is recommended to first configure the CMxCON register with all bits to the desired value, and then set the ON bit. When not used, the Comparator should be disabled expressly by writing a '0' to the ON bit.

## 25.2 Comparator Output Blanking

Comparator output blanking is a feature that is only available on PIC32MK Motor Control (i.e., PIC32MKXXMCXX) devices. The outputs of the Comparators can be further blanked/masked based on external events for programmable durations. This feature can be very useful in reducing the interrupt or trigger frequencies. It is primarily used to select Comparator events (interrupts and triggers) synchronized to desired edge transitions on external digital signals such as the PWM outputs from the MCPWM module. A prudent choice of these external signals has potential to greatly simplify software where otherwise extra software logic will be needed to arbitrate for the desired event source. Refer to the Comparator Mask Control Register, CMxMSKCON (Register 25-3), for details on the 16 different external signals that can be used for masking.

The logic AND, logic OR and multiplexer blocks shown in Figure 25-6 can be visualized as built-in programmable array logic used to reject the unwanted transitions of the comparator output. For each Comparator, the multiplexers A, B, and C can logically AND or OR either the positive or negative levels (edges) of the 16 different external signals. The outputs of the multiplexers can then be ANDed or ORed together with the AND logic outputs of the multiplexers being further capable of selection for positive or negative transitions as shown in the diagram. For a detailed usage of the output blanking feature, refer to Section 39. "Op Amp/Comparator" (DS60001178) of the "PIC32 Family reference Manual".

**FIGURE 25-6: USER PROGRAMMABLE BLANKING FUNCTION DIAGRAM**

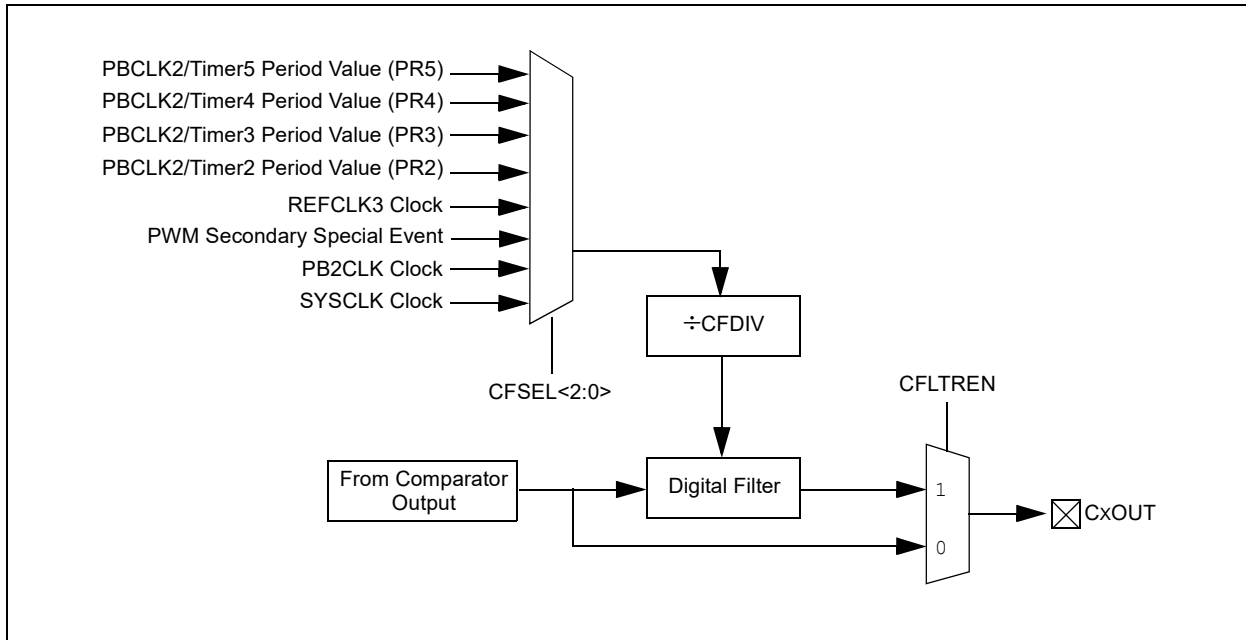


# PIC32MK GPG/MCJ with CAN FD Family

## 25.3 Comparator Output Filtering

The outputs can also be digitally filtered for glitches or noise. The digital filter has the capability to sample at different frequencies using different clock sources specified by the CFSEL<2:0> bits in the CxCON register. The digital filter looks for three consecutive samples of the same logic state before updating the comparator output. Since the digital filter affects the response times of the output, care should be taken while choosing the filter clock divisor to best suit the application at hand.

**FIGURE 25-7: DIGITAL FILTER INTERCONNECT BLOCK DIAGRAM**



# PIC32MK GPG/MCJ with CAN FD Family

## 25.4 Op amp Interface

PIC32MK GP devices implement five comparators and four op amps. The op amp/Comparator module 4 does not implement the associated op amp. The op amp can be configured to operate in two different modes: Regular op amp mode and Unity Gain mode.

When an op amp is available on a op amp/Comparator module, both of its inputs and output are accessible at the device pins. The op amp's Unity Gain mode is the only exception to this rule, which is described in [25.9 "Comparator Configuration"](#). The op amp is disabled at reset and has to be enabled by writing a '1' to CMxCON<OPAON>, (i.e., (CMxCON<31>).

The op amp outputs are capable of rail-to-rail operation, which are limited by the maximum output load current. Refer to [36.0 "Electrical Characteristics"](#) for the op amp minimum gain requirements and VOH/VOL loading specifications.

**Note:** The exception to the minimum gain specification is the special internal Unity Gain buffer mode.

[Table 25-1](#) provides the different SFR bits and their logic states to set the op amp in two different modes of operation.

**TABLE 25-1: OP AMP(X) FUNCTIONAL MODES (X=1-3, 5)**

| Op-Amp Functional Mode                    | CMxCON<OPAON> | CMxCON<ENPGA> | CMxCON<OPLPWR> |
|---|---------------|---------------|----------------|
| High-Performance Mode > 10 MHz, Gain ≥ 2  | 1             | 0             | 0              |
| High-Performance Unity Gain Mode > 10 MHz | 1             | 1             | 0              |
| Low-Power Mode ≤ 10 MHz, Gain ≥ 2         | 1             | 0             | 1              |
| Low-Power Unity Gain Mode ≤ 10 MHz        | 1             | 1             | 1              |
| Op-amp output Tri-Stated                  | 0             | X             | X              |

## 25.5 Op amp Mode

The op amp in the op amp/Comparator module can be enabled by writing a '1' to CMxCON<OPAON>. When configured this way, the output of the Op amp is available at the OAxOUT pin for the external gain/filtering components to be added in the feedback path.

With the proper configuration of the ADC module, the op amp can be configured such that the ADC can directly sample the output of the op amp without the need to route the op amp output to a separate analog input pin (see [Figure 25-16](#)).

CMxCON<ENPGA> = 1 configures the op amp for unity gain 1x mode configuration. For CMxCON<ENPGA> = 0, the minimum Gain ≥ 2 using external resistors as shown in the op amp configuration figures. The RFB in the differential amplifier configuration example must be part of any calculated maximum IOH/LOL load, see [Figure 25-16](#).

- CM2CON2<30> = 1 for Op amp 2
- CM3CON2<30> = 1 for Op amp 3
- CM5CON2<30> = 1 for Op amp 5

Refer to [36.0 "Electrical Characteristics"](#) for the specifications in this mode.

## 25.6 Op amp Unity Gain Mode

Usually the op amps have a minimum gain stable setting as defined in [Table 36-28](#) in [36.0 "Electrical Characteristics"](#). However, there is one exception in that the op amps have an internal 1x gain setting (i.e., the CMxCON<ENPGA> = 1). The mode utilizes only the Non-inverting input pin of the op amp. This configuration needs no external components. The op amps will be placed in a unity gain/follower mode following a software write to these bits:

- CM1CON2<30> = 1 for Op amp 1

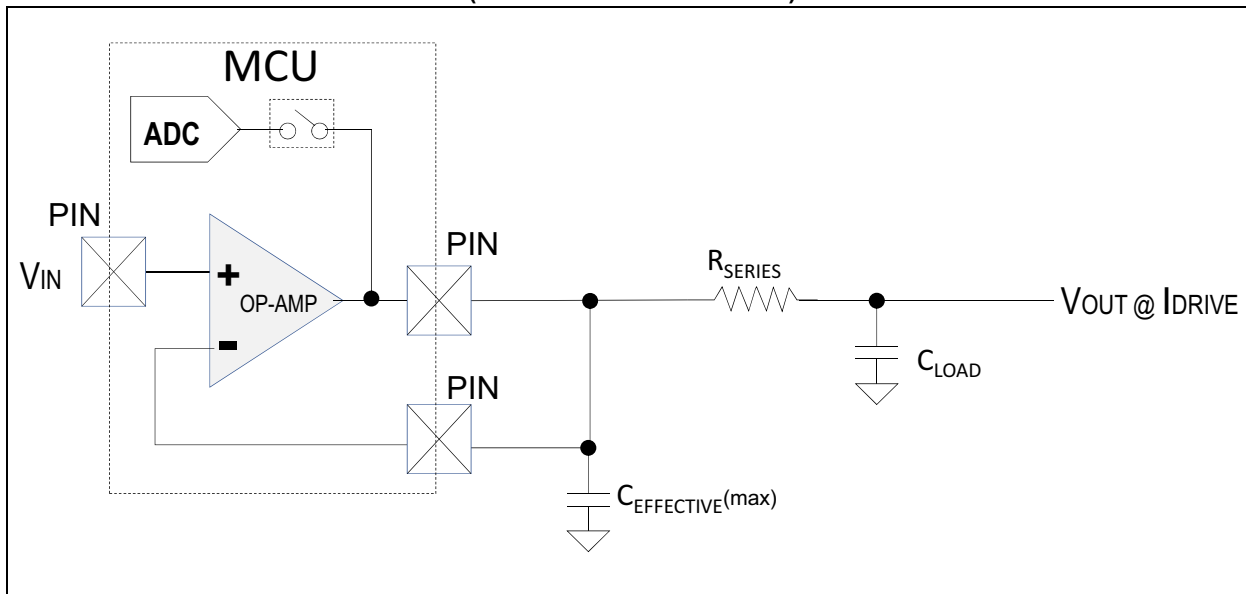
# PIC32MK GPG/MCJ with CAN FD Family

## 25.7 OP-AMP HIGH PERFORMANCE MODE 10MHZ-100MHZ USE REQUIREMENTS (CMxCON<OPLPWR> = 0)

- PIC32 Package I/O pin capacitance/ea. = 4 pf
- Standard PCB = 1.5 pf per 12.5 mm (0.5 inches)  
(TRACE W=0.175 mm, H=36  $\mu$ m, T=113  $\mu$ m)
- $C_{EFFECTIVE}$  = (Package Pin capacitance + PCB trace capacitance)
- $C_{LOAD}$  = (PIC32 Package pin capacitance + PCB trace capacitance + Target destination input pin capacitance)
- BANDWIDTH = User Application Desired Maximum bandwidth  $\leq$  100 MHz
- $V_{IN(max)} = AVDD/Gain$

**Note:** In High-Performance mode  $\geq$  10 MHz all op amp traces must be as short as possible on the same layer if possible as MCU and employ surface mount resistors. Op amp traces must have a continuous ground plane beneath them, and not routed adjacent to digital high frequency signals without a ground guard trace separating them or alternately crossing at right angles to minimize the cross-sectional area for capacitive coupling.

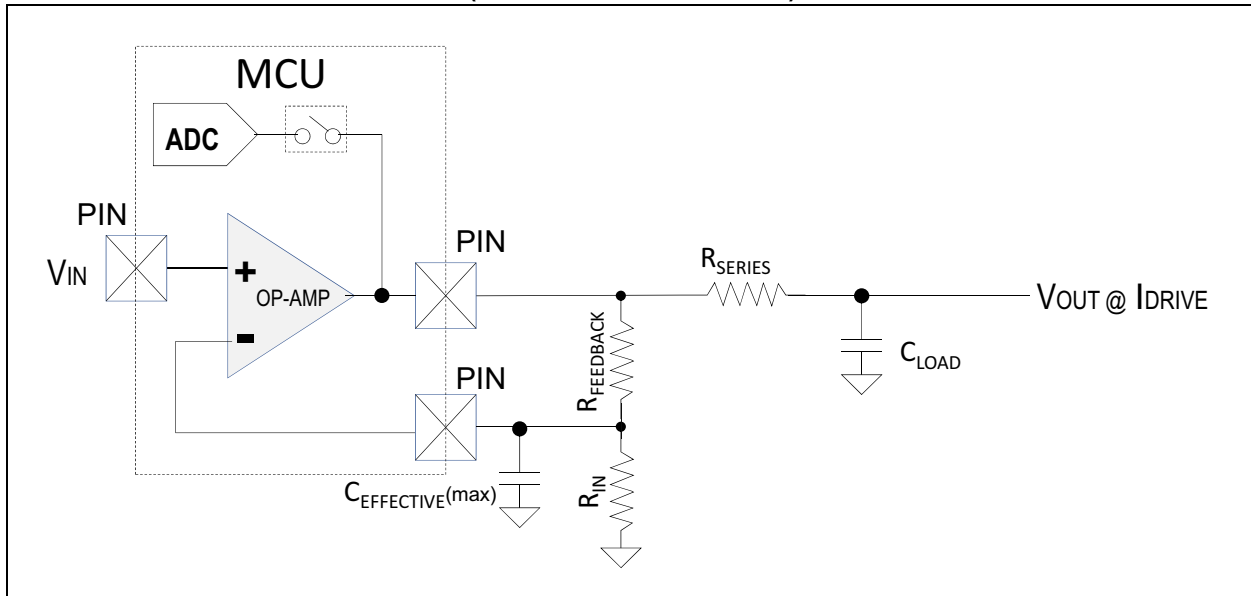
**FIGURE 25-8: OP-AMP UNITY GAIN HIGH PERFORMANCE  $\geq$  10 MHZ SINGLE ENDED MODE REQUIREMENTS (CMXCON<OPLPWR> = 0)**



- $C_{EFFECTIVE} \leq 24$  pf maximum
- $R_{SERIES} = 5 / (2\pi * C_{LOAD} * BANDWIDTH)$
- $V_{OUT} = V_{IN}$

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 25-9: OP-AMP GAIN  $\geq 2$ : HIGH PERFORMANCE  $\geq 10$  MHZ SINGLE ENDED MODE REQUIREMENTS (CMXCON<OPLPWR> = 0)**



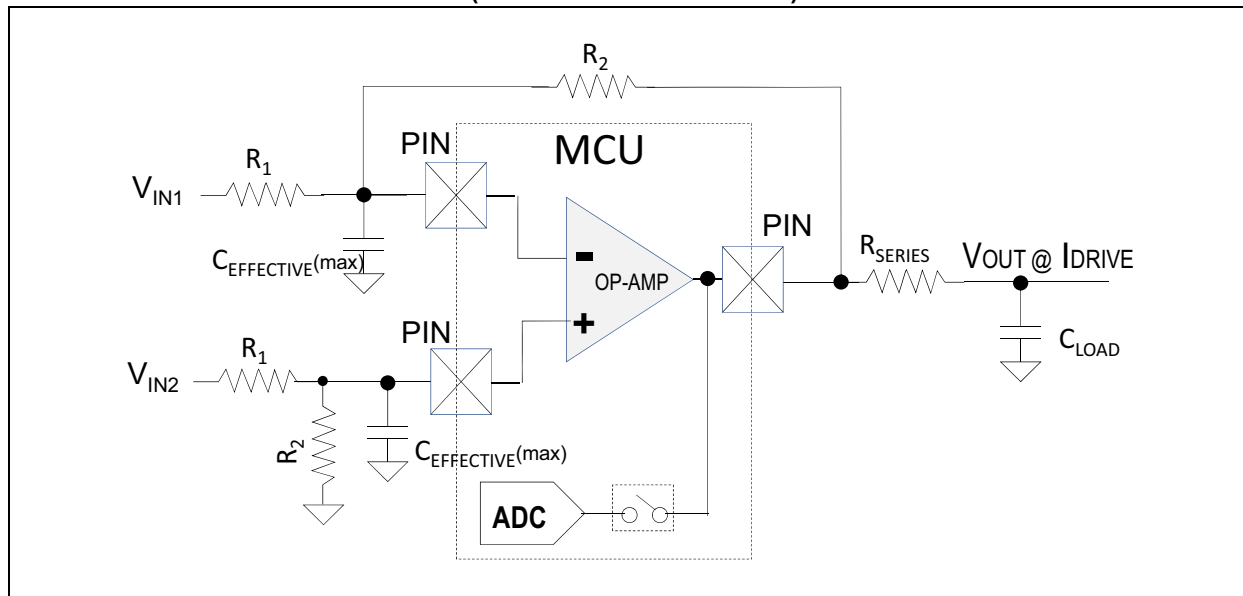
- PIC32 Package I/O pin capacitance/ea. = 4 pf
- Standard PCB = 1.5 pf per 12.5 mm(0.5 inches)  
(TRACE W=0.175 mm, H=36  $\mu$ m, T=113  $\mu$ m)
- CEFFECTIVE = (Package Pin capacitance + PCB trace capacitance)
- CEFFECTIVE  $\leq$  16 pf maximum
- $R_{IN} \Omega = ((8 \text{ pf} / C_{EFFECTIVE}) * 300)$

**Note:** CEFFECTIVE Typ = 8 pf in figure 25-9 if no other node connections.

- $R_{FEEDBACK} \Omega = ((\text{Desired User Gain} - 1) * R_{IN})$
- $\text{Gain} = (1 + (R_{FEEDBACK} / R_{IN}))$
- $I_{DRIVE} (\text{min}) = (10 \text{ ma} - (V_{IN} (\text{pk}) / R_{IN}))$
- $R_{SERIES} = 5 / (2\pi * C_{LOAD} * \text{BANDWIDTH})$
- $V_{OUT} = V_{IN} * \text{Gain}$
- PCB Layout Considerations:
  - RFEEDBACK & RIN should be as close to PIC32 pins as possible
- CLOAD (maximum)  $\leq$  32pf

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 25-10: OP-AMP HIGH PERFORMANCE  $\geq 10$  MHZ DIFFERENTIAL MODE REQUIREMENTS (CMXCON<OPLPWR> = 0)**



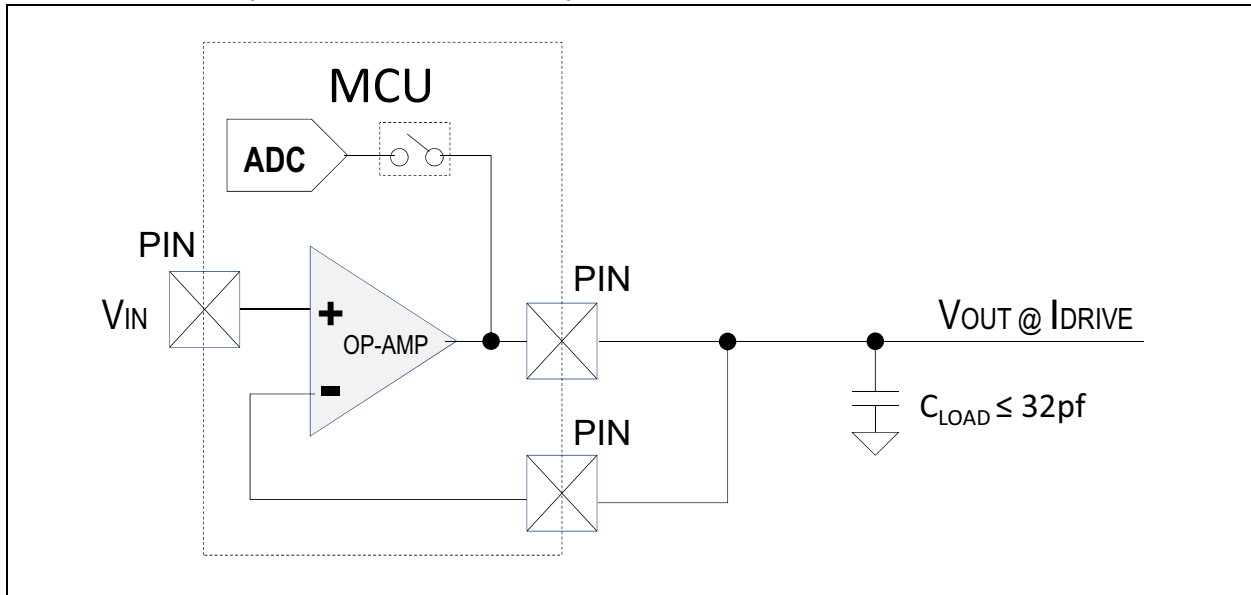
- PIC32 Package I/O pin capacitance/ea. = 4 pf
  - Standard PCB = 1.5 pf per 12.5 mm(0.5 inches)  
(TRACE W=0.175 mm, H=36  $\mu$ m, T=113  $\mu$ m)
  - CEFFECTIVE = (Package Pin capacitance + PCB trace capacitance)
  - CEFFECTIVE  $\leq$  16 pf maximum
  - $R_1 \Omega = ((8 \text{ PF} / \text{CEFFECTIVE}) * 300)$
- Note:** CEFFECTIVE Typ = 8 pf in figure X-3 if no other node connections.
- $R_2 \Omega = ((\text{DESIRED USER GAIN}) * R_1)$
  - $\text{GAIN} = (R_2 / R_1)$
  - $\text{IDRIVE(MIN)} = [10\text{MA} - \{(V_{OUT} / R_2) - (V_{IN2}(\text{PEAK}) / (R_1 + R_2))\}]$
  - $R_{SERIES} = 5 / (2\pi * C_{LOAD} * \text{SIGNAL BANDWIDTH})$
  - $V_{OUT} = (V_{IN2} - V_{IN1}) * \text{GAIN}$
  - PCB LAYOUT CONSIDERATIONS:
    - $R_2$  &  $R_1$  SHOULD BE AS CLOSE TO CPU PINS AS POSSIBLE
  - $C_{LOAD} (\text{max}) \leq 32 \text{ pf}$

# PIC32MK GPG/MCJ with CAN FD Family

## 25.7.1 OP-AMP LOW POWER MODE ≤10MHZ USE REQUIREMENTS (CMXCON<OPLPWR> = 1)

- PIC32 Package I/O pin capacitance/ea. = 4 pf
- Standard PCB = 1.5 pf per 12.5 mm(0.5 inches)  
(TRACE W=0.175 mm, H=36 μm, T=113 μm)
- CEFFECTIVE = (Package Pin capacitance + PCB trace capacitance)
- CLOAD = (CEFFECTIVE + Target destination input pin capacitance)
- VIN(maximum) = AVDD / Gain

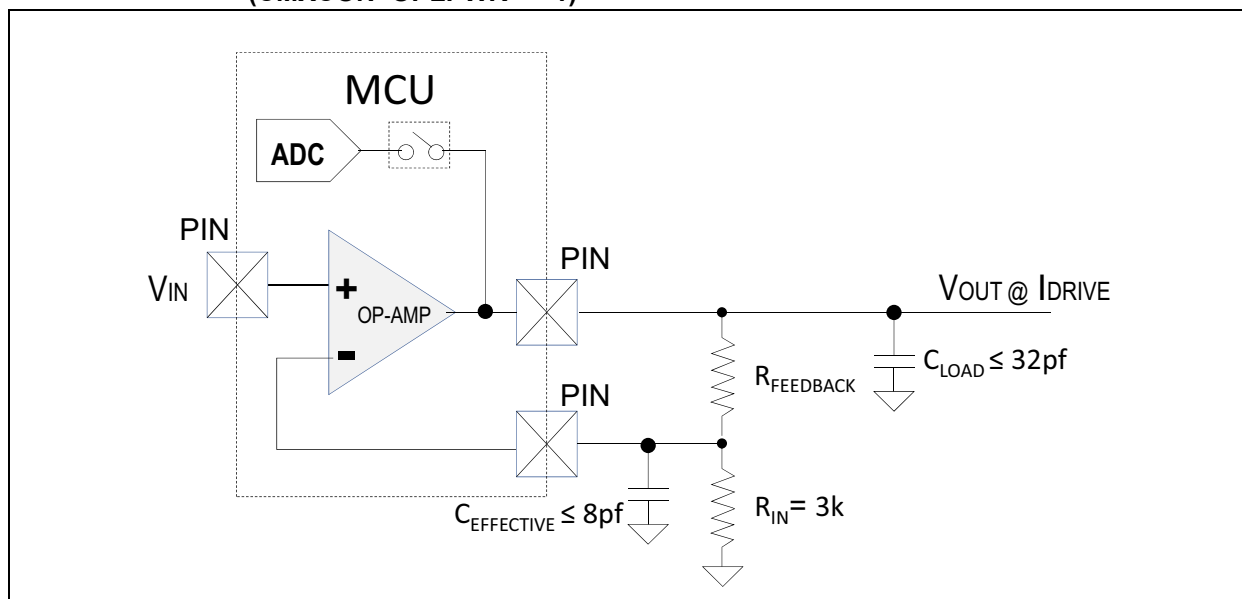
**FIGURE 25-11: OP-AMP UNITY GAIN LOW POWER MODE ≤10MHZ REQUIREMENTS  
(CMXCON<OPLPWR> = 1)**



- CLOAD max ≤ 32 pf
- Standard PCB = 1.5 pf per 12.5 mm(0.5 inches) (TRACE W=0.175 mm, H=36 μm, T=113 μm)

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 25-12: OP-AMP GAIN ≥ 2: LOW POWER MODE ≤10MHZ REQUIREMENTS (CMXCON<OPLPWR> = 1)**

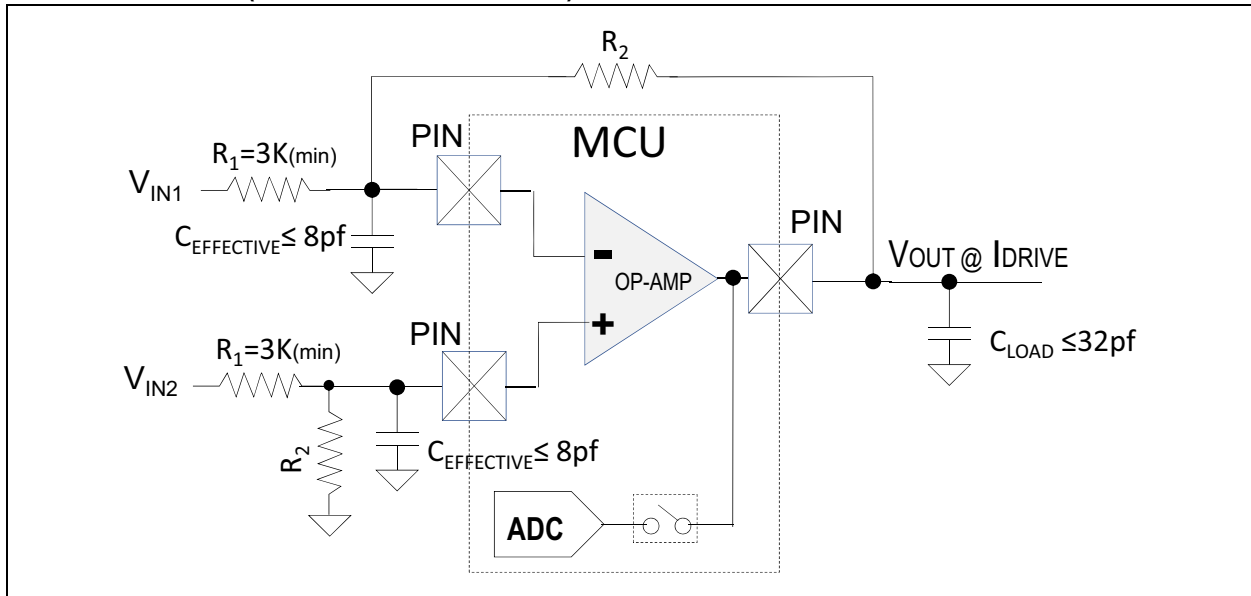


- PIC32 Package I/O pin capacitance/ea. = 4 pf
- Standard PCB = 1.5 pf per 12.5 mm(0.5 inches) (TRACE W=0.175 mm, H=36 μm, T=113 μm)
- CEFFECTIVE = (Package Pin capacitance + PCB trace capacitance)
- Standard PCB = 1.5 pf per 12.5 mm (0.5 inches) (TRACE W=0.175 mm, H=36 μm, T=113 μm)
- CEFFECTIVE max ≤ 8 pf maximum
- RIN = 3k Ω (Highly Recommended)
- RFEEDBACK Ω = ((Desired User Gain – 1) \* RIN)  
= ((Desired User Gain – 1) \* 3k)
- Gain = (1 + (RFEEDBACK / RIN))  
= (1 + (RFEEDBACK / 3k))
- IDRIVE (min) = (10 ma – (VIN (pk) / RIN))  
= (10 ma – (VIN (pk) / 3k))



# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 25-13: OP AMP LOW POWER DIFFERENTIAL MODE  $\leq 10\text{MHz}$  REQUIREMENTS (CMXCON<OPLPWR> = 1)**

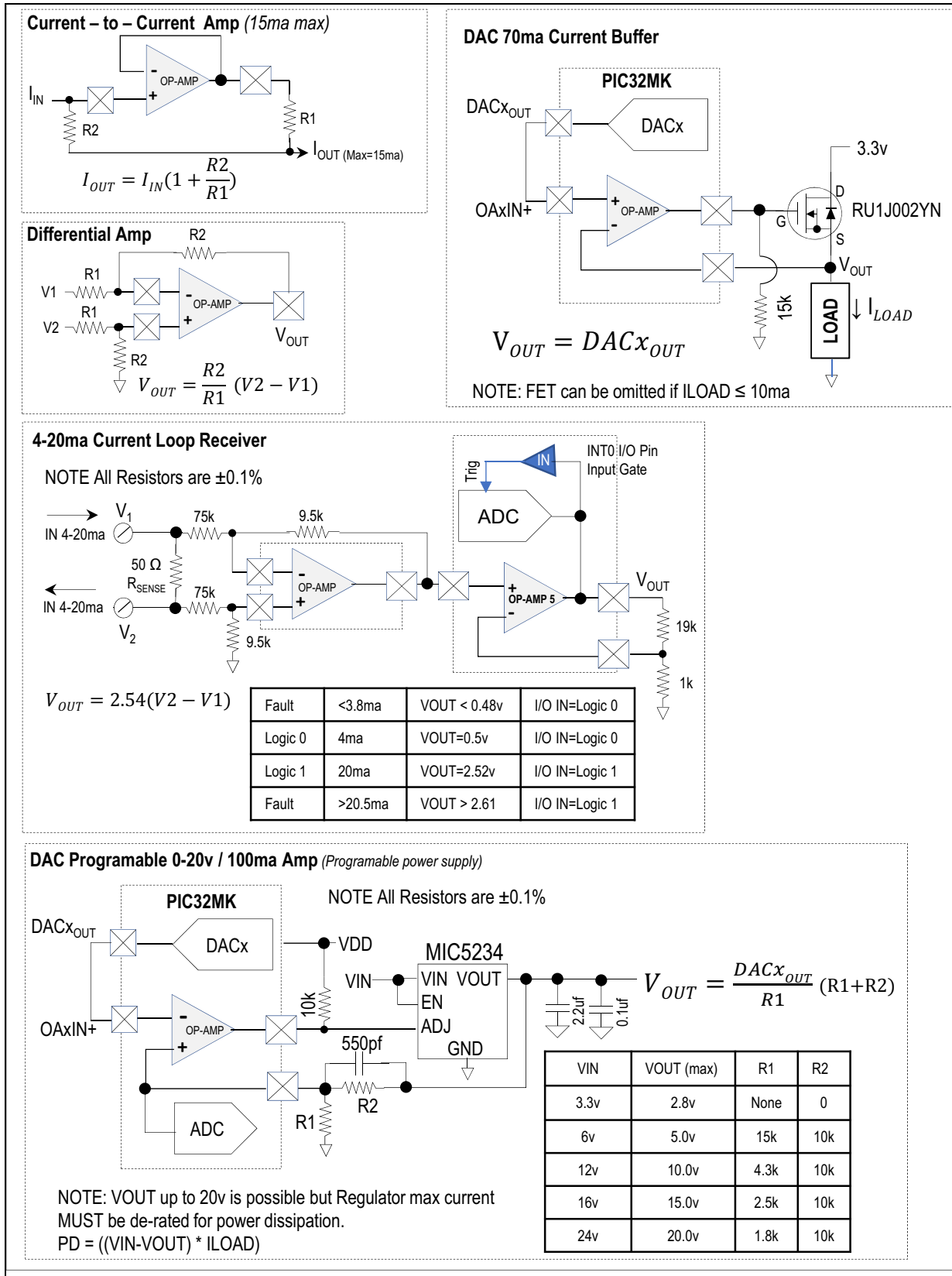


- PIC32 Package I/O pin capacitance/ea. = 4 pf
- Standard PCB = 1.5 pf per 12.5 mm (0.5 inches) (TRACE W=0.175 mm, H=36  $\mu\text{m}$ , T=113  $\mu\text{m}$ )
- CEFFECTIVE = (Package Pin capacitance + PCB trace capacitance)
- Standard PCB = 1.5 pf per 12.5 mm (0.5 inches) (TRACE W=0.175 mm, H=36  $\mu\text{m}$ , T=113  $\mu\text{m}$ )
- CEFFECTIVE max  $\leq 8$  pf maximum
- R1  $\Omega = 3\text{K}$  (min) (Highly Recommended)
- R2  $\Omega = ((\text{Desired User Gain}) * R1)$
- Gain = (R2 / R1)
- IDRIVE(min) = [10 ma - [(VOUT / R2) - (VIN2(peak) / (R1+R2))]]
- VOUT = (VIN2 - VIN1) \* Gain
- PCB Layout Considerations:
  - R2 & R1 should be as close to CPU pins as possible

# PIC32MK GPG/MCJ with CAN FD Family

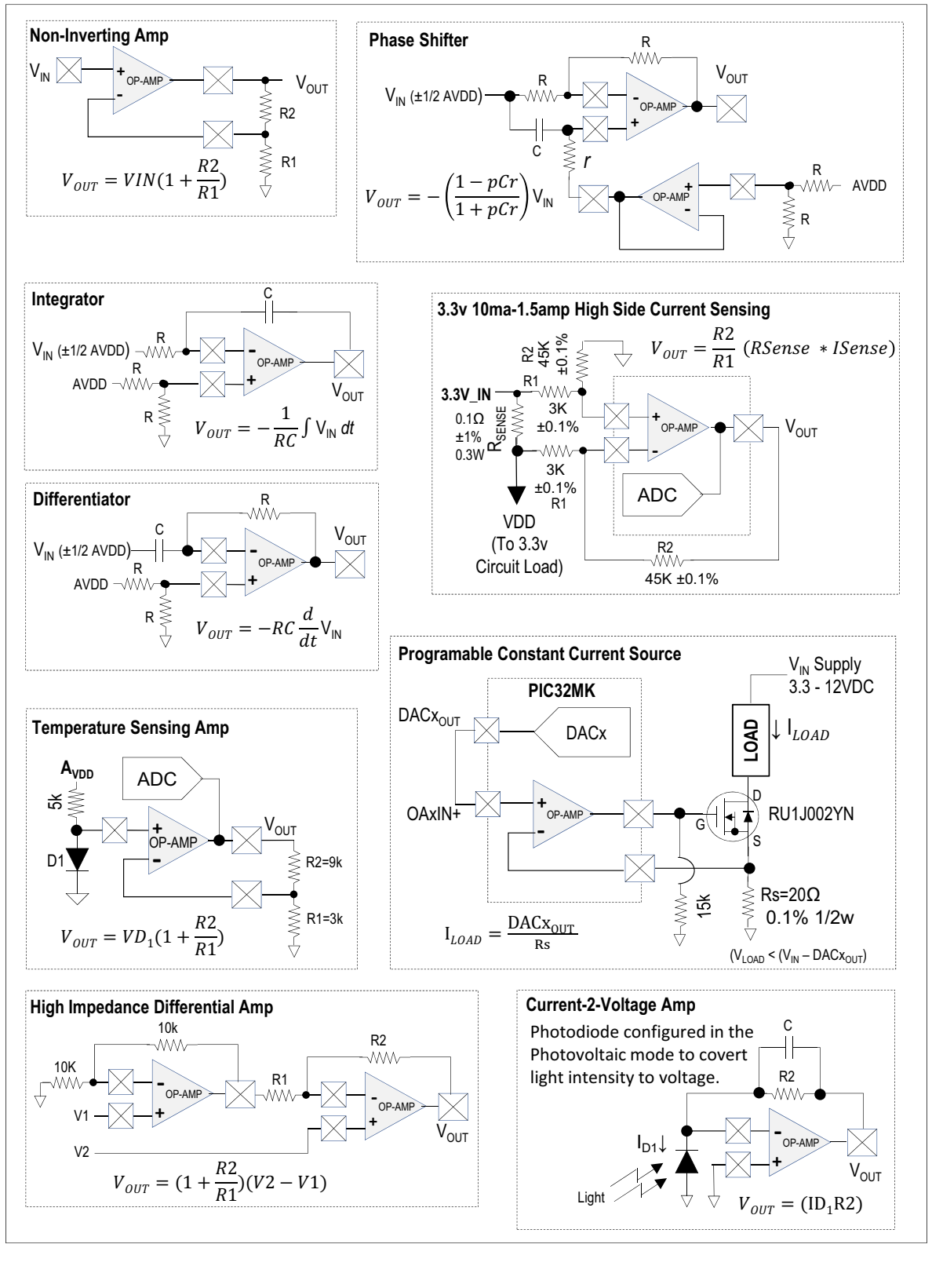
## 25.8 Op-amp Circuit Examples

FIGURE 25-14: OP AMP CONFIGURATION CIRCUIT EXAMPLES



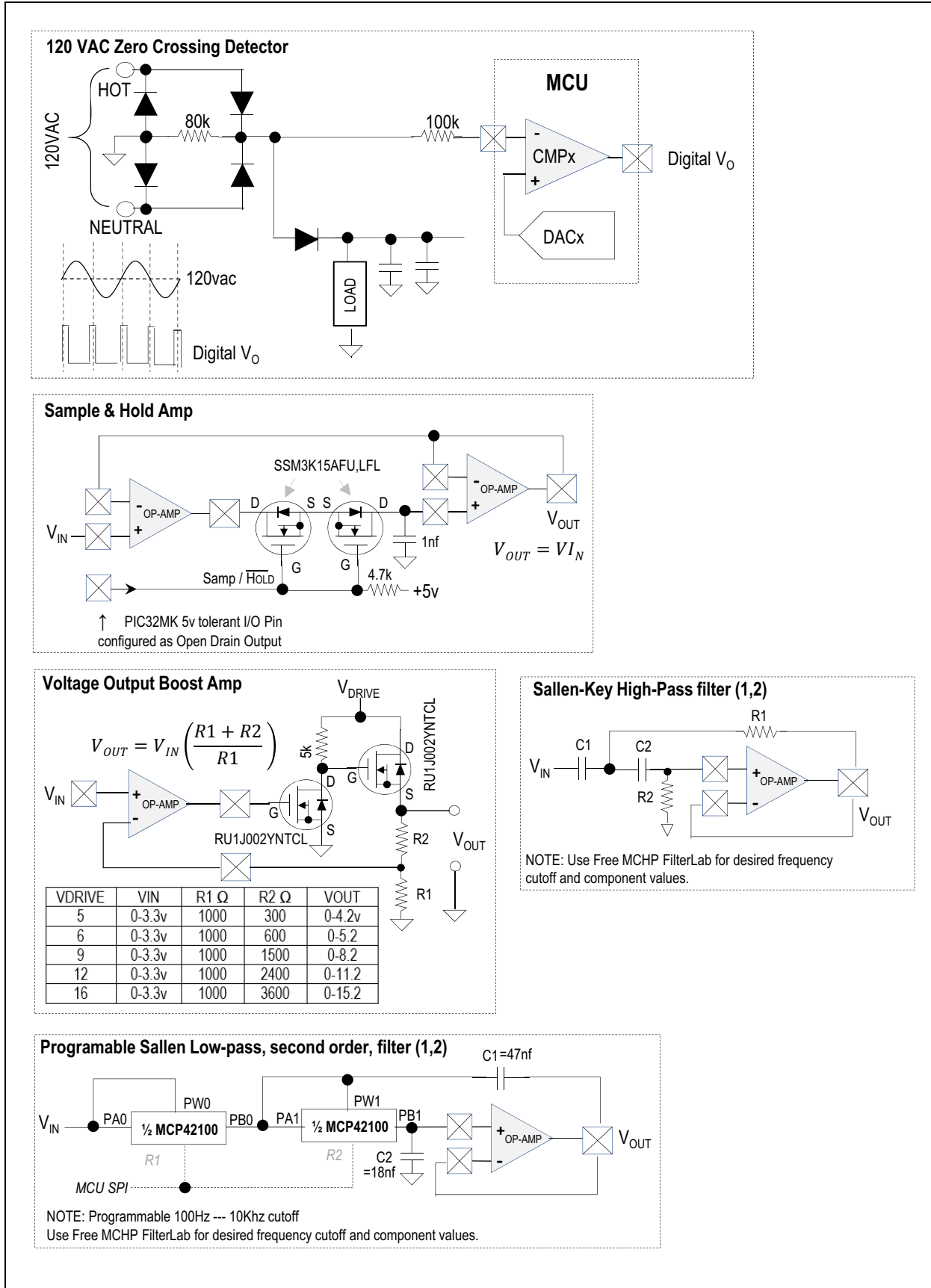
# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 25-15: OP AMP CONFIGURATION CIRCUIT EXAMPLES



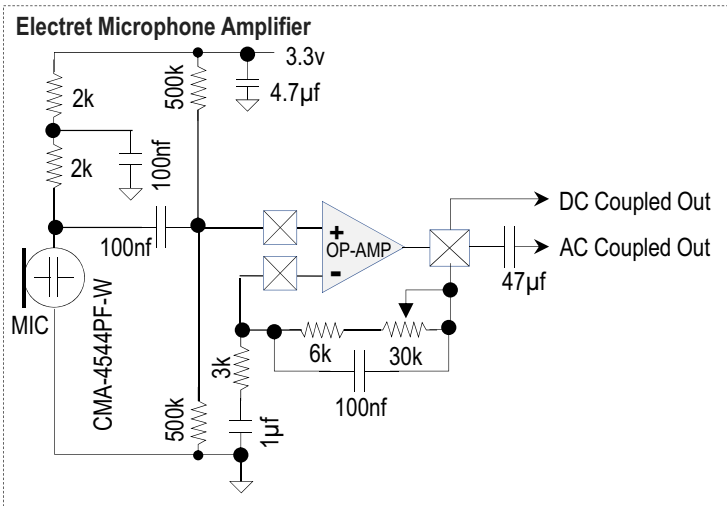
# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 25-16: OP AMP CONFIGURATION CIRCUIT EXAMPLES

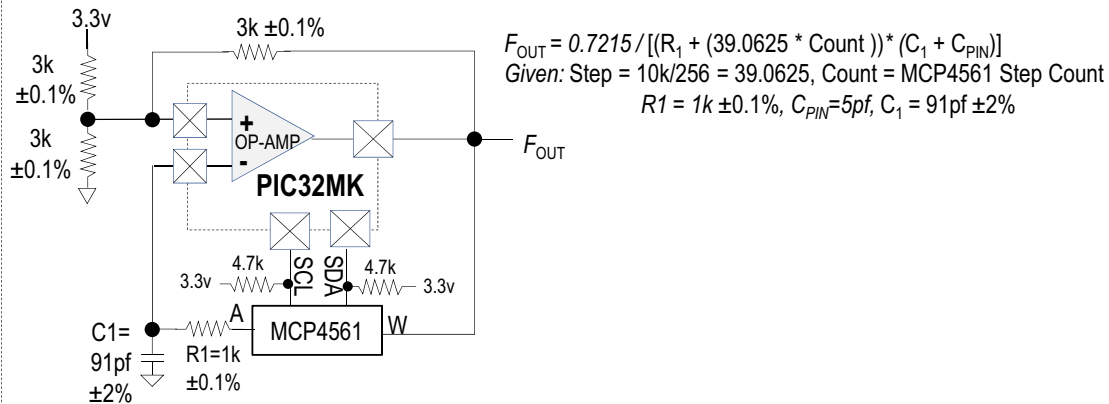


# PIC32MK GPG/MCJ with CAN FD Family

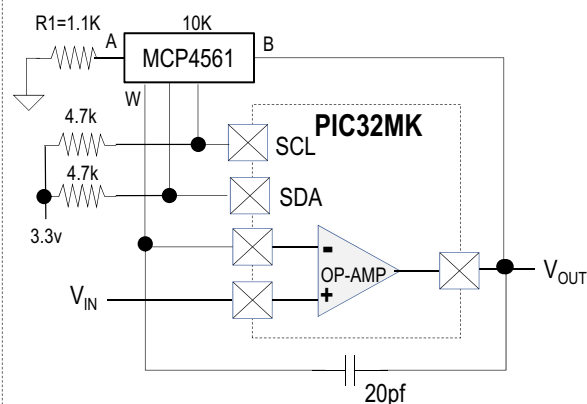
FIGURE 25-17: OP AMP CONFIGURATION CIRCUIT EXAMPLES



**Programmable 7.595Mhz – 690Khz Relaxation Oscillator in 27Khz Steps**

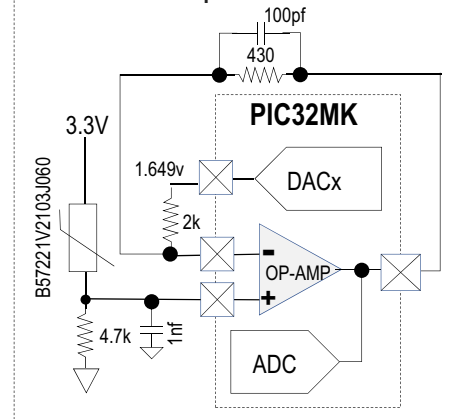


**Programmable Gain Non-Inverting Amp (256 Steps)**



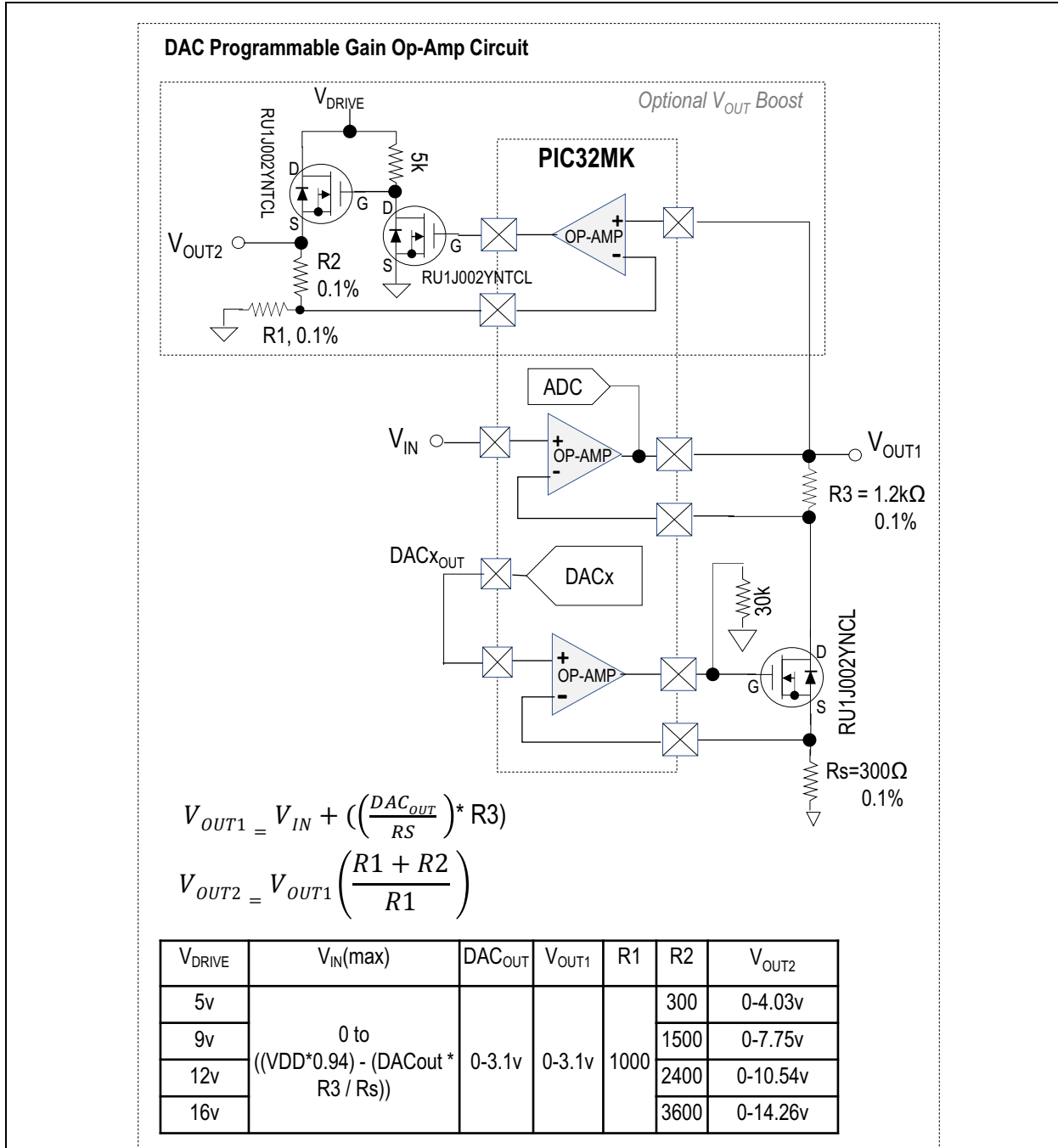
Circuit Gain = 1-10; (Programmable MCP4561 Count)  
 $V_{IN(max)} = (3.1 / \text{Gain(max)})$   
 Step = 10k/256 = 39.0625, Count = MCP4561 Step Count  
 $V_{OUT} = V_{IN} * ((1 + (39.0625 * \text{Count}) / R1 + (10k - (39.0625 * \text{Count})))$

**0-100°C NTC Temperature Sensor**



# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 25-18: OP AMP CONFIGURATION CIRCUIT EXAMPLES



**Note 1:** Microchip's FilterLab<sup>®</sup> software is an innovative software tool that simplifies analog active filter (using op amps) design. Available at no cost from the Microchip web site at [www.microchip.com/filterlab](http://www.microchip.com/filterlab), the FilterLab design tool provides full schematic diagrams of the filter circuit with component values. It also outputs the filter circuit in SPICE format, which can be used with the macro model to simulate actual filter performance.

**2:** Microchip's Mindi<sup>™</sup> simulator tool aids in the design of various circuits useful for active filter, amplifier and power-management applications. It is a free online simulation tool available for download from the Microchip web site at [www.microchip.com/mindi](http://www.microchip.com/mindi). This interactive simulator enables designers to quickly generate circuit diagrams, simulate circuits. Circuits developed using the Mindi simulation tool can be downloaded to a personal computer or workstation.

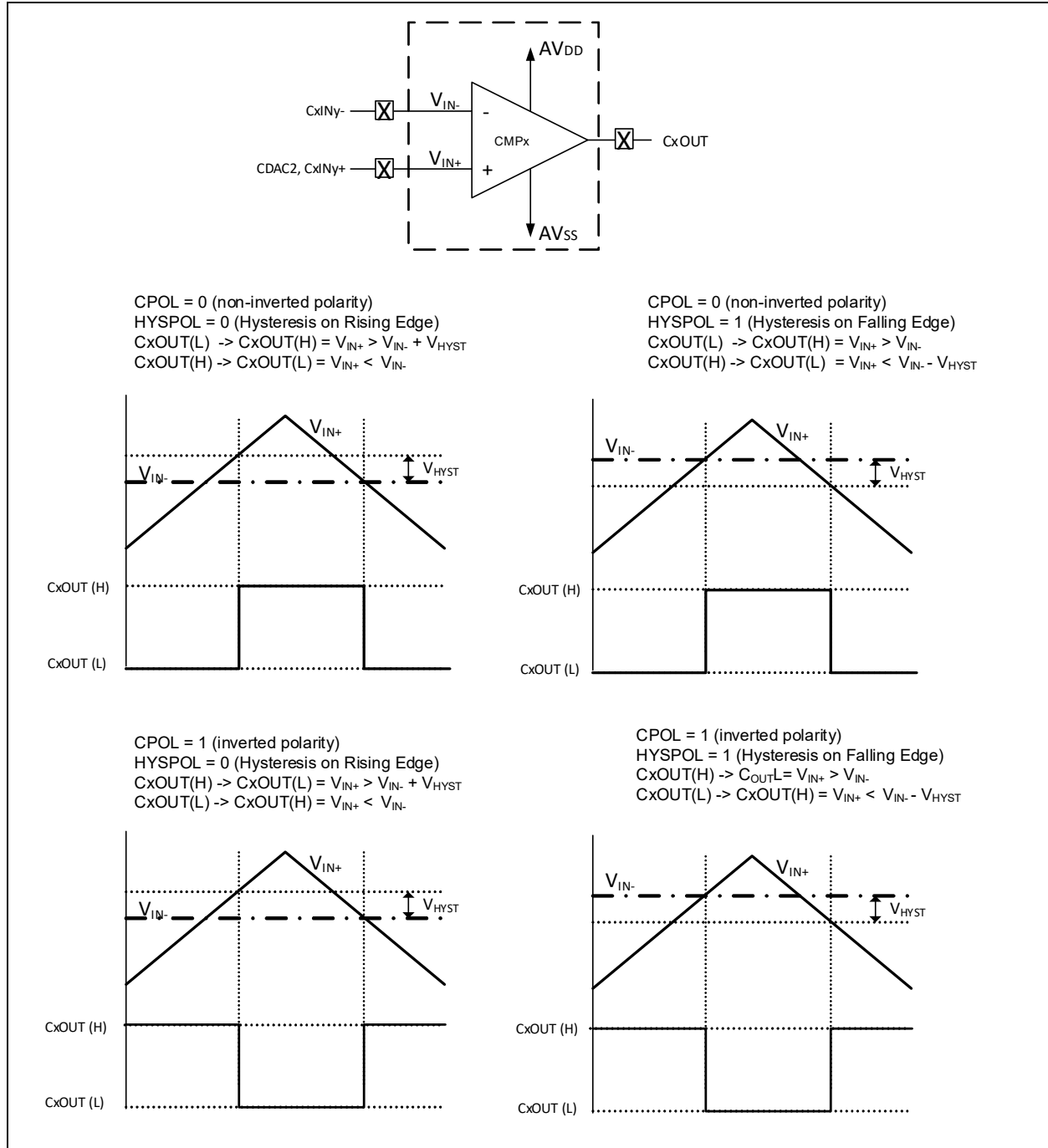
# PIC32MK GPG/MCJ with CAN FD Family

## 25.9 Comparator Configuration

The Comparator and the relationship between the analog input levels and the digital output are illustrated in Figure 25-19. Each Comparator can be individually configured to compare against an external voltage reference or internal voltage reference. For more information on the internal op amp/comparator voltage reference, refer to Section 45. “Control Digital-to-Analog converter” (DS60001327) of the “PIC32 Family Reference Manual”.

A standard configuration with default built in hysteresis is shown in Figure 25-19. The external reference at  $V_{IN+}$  is a fixed voltage. The analog input signal at  $V_{IN-}$  is compared to the reference signal at  $V_{IN+}$ , and the digital output of the comparator is created by the difference between the two signals as shown in the figure. The polarity of the comparator output can be inverted by writing a '1' to the CPOL bit (CMxCON<13>) such that the output is a digital low level when  $V_{IN+} > V_{IN-}$ .

**FIGURE 25-19: COMPARATOR CONFIGURATION FOR DEFAULT BUILT-IN HYSTERESIS**



# PIC32MK GPG/MCJ with CAN FD Family

TABLE 25-2: COMPARATOR CMP<sub>x</sub> OUTPUT AND EVENT POLARITY SELECTION (x=1-5)

| CPOL | EVPOL [1:0] | Comparator Input Change (comparator_status_in) | Comparator Output | CxOUT | Trigger/Interrupt Generated? |
|------|-------------|--|-------------------|-------|------------------------------|
| 0    | 0 0         | +In > -In                                      | High              | High  | No                           |
| 0    | 0 0         | +In < -In                                      | Low               | Low   | No                           |
| 0    | 0 1         | +In > -In                                      | High              | High  | Yes (only while CEVT = 0)    |
| 0    | 0 1         | +In < -In                                      | Low               | Low   | No                           |
| 0    | 1 0         | +In > -In                                      | High              | High  | No                           |
| 0    | 1 0         | +In < -In                                      | Low               | Low   | Yes (only while CEVT = 0)    |
| 0    | 1 1         | +In > -In                                      | High              | High  | Yes (only while CEVT = 0)    |
| 0    | 1 1         | +In < -In                                      | Low               | Low   | Yes (only while CEVT = 0)    |
| 1    | 0 0         | +In > -In                                      | High              | Low   | No                           |
| 1    | 0 0         | +In < -In                                      | Low               | High  | No                           |
| 1    | 0 0         | +In > -In                                      | High              | Low   | No                           |
| 1    | 0 0         | +In < -In                                      | Low               | High  | Yes (only while CEVT = 0)    |
| 1    | 1 0         | +In > -In                                      | High              | Low   | Yes (only while CEVT = 0)    |
| 1    | 1 0         | +In < -In                                      | Low               | High  | No                           |
| 1    | 1 1         | +In > -In                                      | High              | Low   | Yes (only while CEVT = 0)    |
| 1    | 1 1         | +In < -In                                      | Low               | High  | Yes (only while CEVT = 0)    |

**Note:** Each comparator has its dedicated Polarity and Event selection. There is no interaction between the settings of the multiple comparators.



## 25.10 Op amp/Comparator Control Registers

**TABLE 25-3: OP AMP/COMPARATOR REGISTER MAP**

| Virtual Address (BF82) | Register Name <sup>(1)</sup> | Bit Range | Bits  |       |       |        |              |       |             |       |            |              |            |       |       |              |            | All Resets |       |      |
|------------------------|------------------------------|-----------|-------|-------|-------|--------|--------------|-------|-------------|-------|------------|--------------|------------|-------|-------|--------------|------------|------------|-------|------|
|                        |                              |           | 31/15 | 30/14 | 29/13 | 28/12  | 27/11        | 26/10 | 25/9        | 24/8  | 23/7       | 22/6         | 21/5       | 20/4  | 19/3  | 18/2         | 17/1       |            | 16/0  |      |
| C000                   | CMSTAT                       | 31:16     | —     | —     | —     | —      | —            | —     | —           | —     | —          | —            | —          | —     | C5EVT | C4EVT        | C3EVT      | C2EVT      | C1EVT | 0000 |
|                        |                              | 15:0      | —     | —     | SIDL  | —      | —            | —     | —           | —     | —          | —            | —          | —     | C5OUT | C4OUT        | C3OUT      | C2OUT      | C1OUT | 0000 |
| C010                   | CM1CON                       | 31:16     | OPAON | ENPGA | —     | HYSPOL | —            | —     | HYSSEL<1:0> |       |            | —            | CFSEL<2:0> |       |       | CFLTREN      | CFDIV<2:0> |            |       | 0000 |
|                        |                              | 15:0      | ON    | COE   | CPOL  | CLPWR  | —            | —     | CEVT        | COUT  | EVPOL<1:0> |              | —          | CREF  | —     | —            | CCH<1:0>   |            | 0000  |      |
| C020                   | CM1MSKCON                    | 31:16     | —     | —     | —     | —      | SELSRCC<3:0> |       |             |       |            | SELSRCB<3:0> |            |       |       | SELSRCA<3:0> |            |            |       | 0000 |
|                        |                              | 15:0      | HLMS  | —     | OCEN  | OCNEN  | OBEN         | OBNEN | OAEN        | OANEN | NAGS       | PAGS         | ACEN       | ACNEN | ABEN  | ABNEN        | AAEN       | AANEN      | 0000  |      |
| C030                   | CM2CON                       | 31:16     | OPAON | ENPGA | —     | HYSPOL | —            | —     | HYSSEL<1:0> |       |            | —            | CFSEL<2:0> |       |       | CFLTREN      | CFDIV<2:0> |            |       | 0000 |
|                        |                              | 15:0      | ON    | COE   | CPOL  | CLPWR  | —            | —     | CEVT        | COUT  | EVPOL<1:0> |              | —          | CREF  | —     | —            | CCH<1:0>   |            | 0000  |      |
| C040                   | CM2MSKCON                    | 31:16     | —     | —     | —     | —      | SELSRCC<3:0> |       |             |       |            | SELSRCB<3:0> |            |       |       | SELSRCA<3:0> |            |            |       | 0000 |
|                        |                              | 15:0      | HLMS  | —     | OCEN  | OCNEN  | OBEN         | OBNEN | OAEN        | OANEN | NAGS       | PAGS         | ACEN       | ACNEN | ABEN  | ABNEN        | AAEN       | AANEN      | 0000  |      |
| C050                   | CM3CON                       | 31:16     | OPAON | ENPGA | —     | HYSPOL | —            | —     | HYSSEL<1:0> |       |            | —            | CFSEL<2:0> |       |       | CFLTREN      | CFDIV<2:0> |            |       | 0000 |
|                        |                              | 15:0      | ON    | COE   | CPOL  | CLPWR  | —            | —     | CEVT        | COUT  | EVPOL<1:0> |              | —          | CREF  | —     | —            | CCH<1:0>   |            | 0000  |      |
| C060                   | CM3MSKCON                    | 31:16     | —     | —     | —     | —      | SELSRCC<3:0> |       |             |       |            | SELSRCB<3:0> |            |       |       | SELSRCA<3:0> |            |            |       | 0000 |
|                        |                              | 15:0      | HLMS  | —     | OCEN  | OCNEN  | OBEN         | OBNEN | OAEN        | OANEN | NAGS       | PAGS         | ACEN       | ACNEN | ABEN  | ABNEN        | AAEN       | AANEN      | 0000  |      |
| C070                   | CM4CON                       | 31:16     | —     | —     | —     | HYSPOL | —            | —     | HYSSEL<1:0> |       |            | —            | CFSEL<2:0> |       |       | CFLTREN      | CFDIV<2:0> |            |       | 0000 |
|                        |                              | 15:0      | ON    | COE   | CPOL  | —      | —            | —     | CEVT        | COUT  | EVPOL<1:0> |              | —          | CREF  | —     | —            | CCH<1:0>   |            | 0000  |      |
| C080                   | CM4MSKCON                    | 31:16     | —     | —     | —     | —      | SELSRCC<3:0> |       |             |       |            | SELSRCB<3:0> |            |       |       | SELSRCA<3:0> |            |            |       | 0000 |
|                        |                              | 15:0      | HLMS  | —     | OCEN  | OCNEN  | OBEN         | OBNEN | OAEN        | OANEN | NAGS       | PAGS         | ACEN       | ACNEN | ABEN  | ABNEN        | AAEN       | AANEN      | 0000  |      |
| C090                   | CM5CON                       | 31:16     | OPAON | ENPGA | —     | HYSPOL | —            | —     | HYSSEL<1:0> |       |            | —            | CFSEL<2:0> |       |       | CFLTREN      | CFDIV<2:0> |            |       | 0000 |
|                        |                              | 15:0      | ON    | COE   | CPOL  | CLPWR  | —            | —     | CEVT        | COUT  | EVPOL<1:0> |              | —          | CREF  | —     | —            | CCH<1:0>   |            | 0000  |      |
| C0A0                   | CM5MSKCON                    | 31:16     | —     | —     | —     | —      | SELSRCC<3:0> |       |             |       |            | SELSRCB<3:0> |            |       |       | SELSRCA<3:0> |            |            |       | 0000 |
|                        |                              | 15:0      | HLMS  | —     | OCEN  | OCNEN  | OBEN         | OBNEN | OAEN        | OANEN | NAGS       | PAGS         | ACEN       | ACNEN | ABEN  | ABNEN        | AAEN       | AANEN      | 0000  |      |

**Legend:** \* = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 25-1: CMSTAT: OP AMP/COMPARATOR STATUS REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0            | U-0            | U-0            | HS, R-0        | HS, R-0        | HS, R-0        | HS, R-0       | HS, R-0       |
|           | —              | —              | —              | C5EVT          | C4EVT          | C3EVT          | C2EVT         | C1EVT         |
| 15:8      | U-0            | U-x            | R/W-0          | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —              | —              | SIDL           | —              | —              | —              | —             | —             |
| 7:0       | U-0            | U-0            | U-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | —              | —              | —              | C5OUT          | C4OUT          | C3OUT          | C2OUT         | C1OUT         |

|                   |  |
|-------------------|--|
| <b>Legend:</b>    | HS = Set by hardware   |
| R = Readable bit  | W = Writable bit      U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown |

- bit 31-21 **Unimplemented:** Read as '0'
- bit 20 **C5EVT:** Comparator 5 Event Status bit
  - 1 = Comparator event according to EVPOL<1:0> settings occurred. Future events/triggers and interrupts are disabled until the CEVT bit (CMxCON<9>) is cleared by user software.
  - 0 = Comparator event did not occur
- bit 19 **C4EVT:** Comparator 4 Event Status bit
  - 1 = Comparator event according to EVPOL<1:0> settings occurred. Future events/triggers and interrupts are disabled until the CEVT bit (CMxCON<9>) is cleared by user software.
  - 0 = Comparator event did not occur
- bit 18 **C3EVT:** Comparator 3 Event Status bit
  - 1 = Comparator event according to EVPOL<1:0> settings occurred. Future events/triggers and interrupts are disabled until the CEVT bit (CMxCON<9>) is cleared by user software.
  - 0 = Comparator event did not occur
- bit 17 **C2EVT:** Comparator 2 Event Status bit
  - 1 = Comparator event according to EVPOL<1:0> settings occurred. Future events/triggers and interrupts are disabled until the CEVT bit (CMxCON<9>) is cleared by user software.
  - 0 = Comparator event did not occur
- bit 16 **C1EVT:** Comparator 1 Event Status bit
  - 1 = Comparator event according to EVPOL<1:0> settings occurred. Future events/triggers and interrupts are disabled until the CEVT bit (CMxCON<9>) is cleared by user software.
  - 0 = Comparator event did not occur
- bit 15-14 **Unimplemented:** Read as '0'
- bit 13 **SIDL:** Stop in Idle Mode bit
  - 1 = Discontinue operation of all Op amp/Comparators when device enters Idle mode
  - 0 = Continue module operation in Idle mode
- bit 12-5 **Unimplemented:** Read as '0'
- bit 4-0 **C5OUT:C1OUT:** Op amp/Comparator 5 through Comparator 1 Output Status bit
  - When CPOL = 0:
    - 1 =  $V_{IN+} > V_{TH+}$
    - 0 =  $V_{IN+} < V_{TH-}$
  - When CPOL = 1:
    - 1 =  $V_{IN+} < V_{TH-}$
    - 0 =  $V_{IN+} > V_{TH+}$

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 25-2: CMxCON: OP AMP/COMPARATOR 'x' CONTROL REGISTER**  
(‘x’ = 1-5)

| Bit Range | Bit 31/23/15/7       | Bit 30/22/14/6       | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1       | Bit 24/16/8/0 |
|-----------|----------------------|----------------------|----------------|----------------|----------------|----------------|---------------------|---------------|
| 31:24     | R/W-0                | R/W-0                | U-0            | R/W-0          | U-0            | U-0            | R/W-0               | R/W-0         |
|           | OPAON <sup>(2)</sup> | ENPGA <sup>(2)</sup> | —              | HYSPOL         | —              | —              | HYSSEL<1:0>         |               |
| 23:16     | U-0                  | R/W-0                | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0         |
|           | —                    | CFSEL<2:0>           |                |                | CFLTREN        | CFDIV<2:0>     |                     |               |
| 15:8      | R/W-0                | R/W-0                | R/W-0          | R/W-0          | U-0            | U-0            | HS, R/W-0           | R-0           |
|           | ON                   | COE                  | CPOL           | OPLPWR         | —              | —              | CVET <sup>(1)</sup> | COUT          |
| 7:0       | R/W-0                | R/W-0                | U-0            | R/W-0          | U-0            | U-0            | R/W-0               | R/W-0         |
|           | EVPOL<1:0>           |                      | —              | CREF           | —              | —              | CCH<1:0>            |               |

|                   |                                    |
|-------------------|------------------------------------|
| <b>Legend:</b>    | HS = Set by hardware               |
| R = Readable bit  | W = Writable bit                   |
| -n = Value at POR | U = Unimplemented bit, read as '0' |
|                   | '1' = Bit is set                   |
|                   | '0' = Bit is cleared               |
|                   | x = Bit is unknown                 |

bit 31 **OPAON:** Op amp Enable bit<sup>(2)</sup>

- 1 = Op amp enabled and connected to pin
- 0 = Op amp disabled and pin is released to other functions

**Note:** Initialize ENPGA & OPLPWR prior to enabling the op-amp, CMxCON<OPAON>=1.

bit 30 **ENPGA:** Op amp Fixed Gain Enable bit<sup>(2)</sup>

- 1 = Op amp is operating in Fixed Gain 1X mode
- 0 = Op amp is operating in Open Loop mode (default)

bit 29 **Unimplemented:** Read as '0'

bit 28 **HYSPOL:** Comparator Hysteresis Polarity Selection

- 1 = Hysteresis on falling edge, rising edge is accurate
- 0 = Hysteresis on rising edge, falling edge is accurate

bit 27-26 **Unimplemented:** Read as '0'

bit 25-24 **HYSSEL<1:0>:** Hysteresis Selection bits

- 11 = Set highest hysteresis level (Typical 45 mV)
- 10 = Set medium hysteresis level (Typical 30 mV)
- 01 = Set lowest hysteresis level (Typical 15 mV)
- 00 = No hysteresis selected.

**Note:** These bits select the hysteresis of the analog comparator.

bit 23 **Unimplemented:** Read as '0'

bit 22-20 **CFSEL<2:0>:** Comparator Output Filter Clock Source Select bits

- 111 = PBCLK2/Timer5 Period Value (PR5)
- 110 = PBCLK2/Timer4 Period Value (PR4)
- 101 = PBCLK2/Timer3 Period Value (PR3)
- 100 = PBCLK2/Timer2 Period Value (PR2)
- 011 = REFCLK3 Clock
- 010 = PWM Secondary Special Event
- 001 = PPBCLK2 Clock
- 000 = SYSCLK Clock

**Note 1:** Before attempting to initialize or enable any of the op amp bits, the user application must clear the corresponding OPA5MD, OPA3MD, OPA2MD, OPA1MD bits in the PMD register.

**2:** These bits are not available in the CM4CON register.

**Note:** The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition has occurred. The IFSx bits are persistent, so they must be cleared by user software.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 25-2: CMxCON: OP AMP/COMPARATOR 'x' CONTROL REGISTER ('x' = 1-5) (CONTINUED)

bit 19 **CFLTREN**: Comparator Output Digital Filter Enable bit

- 1 = Digital Filters enabled
- 0 = Digital Filters disabled

bit 18-16 **CFDIV<2:0>**: Comparator Output Filter Clock Divide Select bits

These bits are based on the CFSEL clock source selection.

- 111 = 1:128 Clock Divide
- 110 = 1:64 Clock Divide
- 101 = 1:32 Clock Divide
- 100 = 1:16 Clock Divide
- 011 = 1:8 Clock Divide
- 010 = 1:4 Clock Divide
- 001 = 1:2 Clock Divide
- 000 = 1:1 Clock Divide

bit 15 **ON**: Comparator Enable bit

- 1 = Comparator is enabled
- 0 = Comparator is disabled

bit 14 **COE**: Comparator Output Enable bit

- 1 = Comparator output is present on the CxOUT pin
- 0 = Comparator output is internal only

bit 13 **CPOL**: Comparator Output Polarity Select bit

- 1 = Comparator output is inverted
- 0 = Comparator output is not inverted

bit 12 **OPLPWR**: Op Amp Power mode

- 1 = Op amp operating in low-power mode (1/10 power, slower response, Bandwidth  $\leq$  10 MHz)
- 0 = Op amp operating in normal power mode, (10 MHz  $\geq$  Bandwidth  $\leq$  100 MHz)

**Note:** This bit does not exist for in CM4CON as there is no Op-Amp #4.

bit 11-10 **Unimplemented**: Read as '0'

bit 9 **CEVT**: Comparator Event bit<sup>(1)</sup>

- 1 = Comparator event according to EVPOL<1:0> bit settings occurred

**Note:** CEVT = 1 disables future events/triggers and interrupts until the bit is cleared by user software.

- 0 = Comparator event did not occur

bit 8 **COUT**: Comparator Output bit

When CPOL = 0 (non-inverted polarity):

- 1 =  $V_{IN+} > V_{TH+}$
- 0 =  $V_{IN+} < V_{TH-}$

When CPOL = 1 (inverted polarity):

- 1 =  $V_{IN+} < V_{TH-}$
- 0 =  $V_{IN+} > V_{TH+}$

**Note 1:** Before attempting to initialize or enable any of the op amp bits, the user application must clear the corresponding OPA5MD, OPA3MD, OPA2MD, OPA1MD bits in the PMD register.

**2:** These bits are not available in the CM4CON register.

**Note:** The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition has occurred. The IFSx bits are persistent, so they must be cleared by user software.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 25-2: CMxCON: OP AMP/COMPARATOR 'x' CONTROL REGISTER ( 'x' = 1-5) (CONTINUED)

- bit 7-6 **EVPOL<1:0>**: Trigger/Event Polarity Select bits
- 11 = Trigger/Event generated on any change of the comparator output
  - 10 = Trigger/Event generated only on high-to-low transition of the polarity-selected comparator output
    - If CPOL = 0 (non-inverted polarity):  
High-to-low transition of the comparator output
    - If CPOL = 1 (inverted polarity):  
Low-to-high transition of the comparator output
  - 01 = Trigger/Event generated only on low-to-high transition of the polarity-selected comparator output
    - If CPOL = 0 (non-inverted polarity):  
Low-to-high transition of the comparator output
    - If CPOL = 1 (inverted polarity):  
High-to-low transition of the comparator output
  - 00 = Trigger/Event generation is disabled
- bit 5 **Unimplemented**: Read as '0'
- bit 4 **CREF**: Op amp/Comparator Reference Select bit
- 1 = VIN+ input connects to internal CDAC2 output voltage
  - 0 = VIN+ input connects to CxIN1+ pin
- bit 3-2 **Unimplemented**: Read as '0'
- bit 1-0 **CCH<1:0>**: Comparator Channel Select bits
- 11 = CxIN4-
  - 10 = CxIN3-
  - 01 = CxIN2-
  - 00 = CxIN1-

- Note 1:** Before attempting to initialize or enable any of the op amp bits, the user application must clear the corresponding OPA5MD, OPA3MD, OPA2MD, OPA1MD bits in the PMD register.
- 2:** These bits are not available in the CM4CON register.

**Note:** The IFSx bits, as with all interrupt flag status register bits, are set as long as the peripheral is enabled and an interrupt condition event occurs. Interrupts do not have to be enabled for the IFSx bits to be set. If the user application does not want to use an interrupt, it can poll the corresponding peripheral IFSx bit to see whether an interrupt condition has occurred. The IFSx bits are persistent, so they must be cleared by user software.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 25-3: CMxMSKCON: COMPARATOR 'x' MASK CONTROL REGISTER (‘x’ = 1-5) (CONTINUED)

|        |   |
|--------|---|
| bit 12 | <b>OCNEN:</b> OR Gate “C” Input Inverted Enable bit<br>1 = “C” input (inverted) enabled as input to OR gate<br>0 = “C” input (inverted) disabled as input to OR gate  |
| bit 11 | <b>OBEN:</b> OR Gate “B” Input Enable bit<br>1 = “B” input enabled as input to OR gate<br>0 = “B” input disabled as input to OR gate  |
| bit 10 | <b>OBNEN:</b> OR Gate “B” Input Inverted Enable bit<br>1 = “B” input (inverted) enabled as input to OR gate<br>0 = “B” input (inverted) disabled as input to OR gate  |
| bit 9  | <b>OAEN:</b> OR Gate “A” Input Enable bit<br>1 = “A” input enabled as input to OR gate<br>0 = “A” input disabled as input to OR gate  |
| bit 8  | <b>OANEN:</b> OR Gate “A” Input Inverted Enable bit<br>1 = “A” input (inverted) enabled as input to OR gate<br>0 = “A” input (inverted) disabled as input to OR gate  |
| bit 7  | <b>NAGS:</b> Negative AND Gate Output Select bit<br>1 = The negative (inverted) output of the AND gate to the OR gate is enabled<br>0 = The negative (inverted) output of the AND gate to the OR gate is disabled |
| bit 6  | <b>PAGS:</b> Positive AND Gate Output Select bit<br>1 = The positive output of the AND gate to the OR gate is enabled<br>0 = The positive output of the AND gate to the OR gate is disabled                       |
| bit 5  | <b>ACEN:</b> AND Gate “C” Input Enable bit<br>1 = “C” input enabled as input to AND gate<br>0 = “C” input disabled as input to AND gate   |
| bit 4  | <b>ACNEN:</b> AND Gate “C” Inverted Input Enable bit<br>1 = “C” input (inverted) enabled as input to AND gate<br>0 = “C” input (inverted) disabled as input to AND gate   |
| bit 3  | <b>ABEN:</b> AND Gate “B” Input Enable bit<br>1 = “B” input enabled as input to AND gate<br>0 = “B” input disabled as input to AND gate   |
| bit 2  | <b>ABNEN:</b> AND Gate “B” Inverted Input Enable bit<br>1 = “B” input (inverted) enabled as input to AND gate<br>0 = “B” input (inverted) disabled as input to AND gate   |
| bit 1  | <b>AAEN:</b> AND Gate “A” Input Enable bit<br>1 = “A” input enabled as input to AND gate<br>0 = “A” input disabled as input to AND gate   |
| bit 0  | <b>AANEN:</b> AND Gate “A” Inverted Input Enable bit<br>1 = “A” input (inverted) enabled as input to AND gate<br>0 = “A” input (inverted) disabled as input to AND gate   |

**Note:** This register is only available on PIC32MKXXMCXXX devices.

# PIC32MK GPG/MCJ with CAN FD Family

## 26.0 CHARGE TIME MEASUREMENT UNIT (CTMU)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 37. “Charge Time Measurement Unit (CTMU)”** (DS60001167), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The Charge Time Measurement Unit (CTMU) is a flexible analog module that has a configurable current source with a digital configuration circuit built around it. The CTMU can be used for differential time measurement between pulse sources and can be used for generating an asynchronous pulse. By working with other on-chip analog modules, the CTMU can be used for high resolution time measurement, measure capacitance, measure relative changes in capacitance

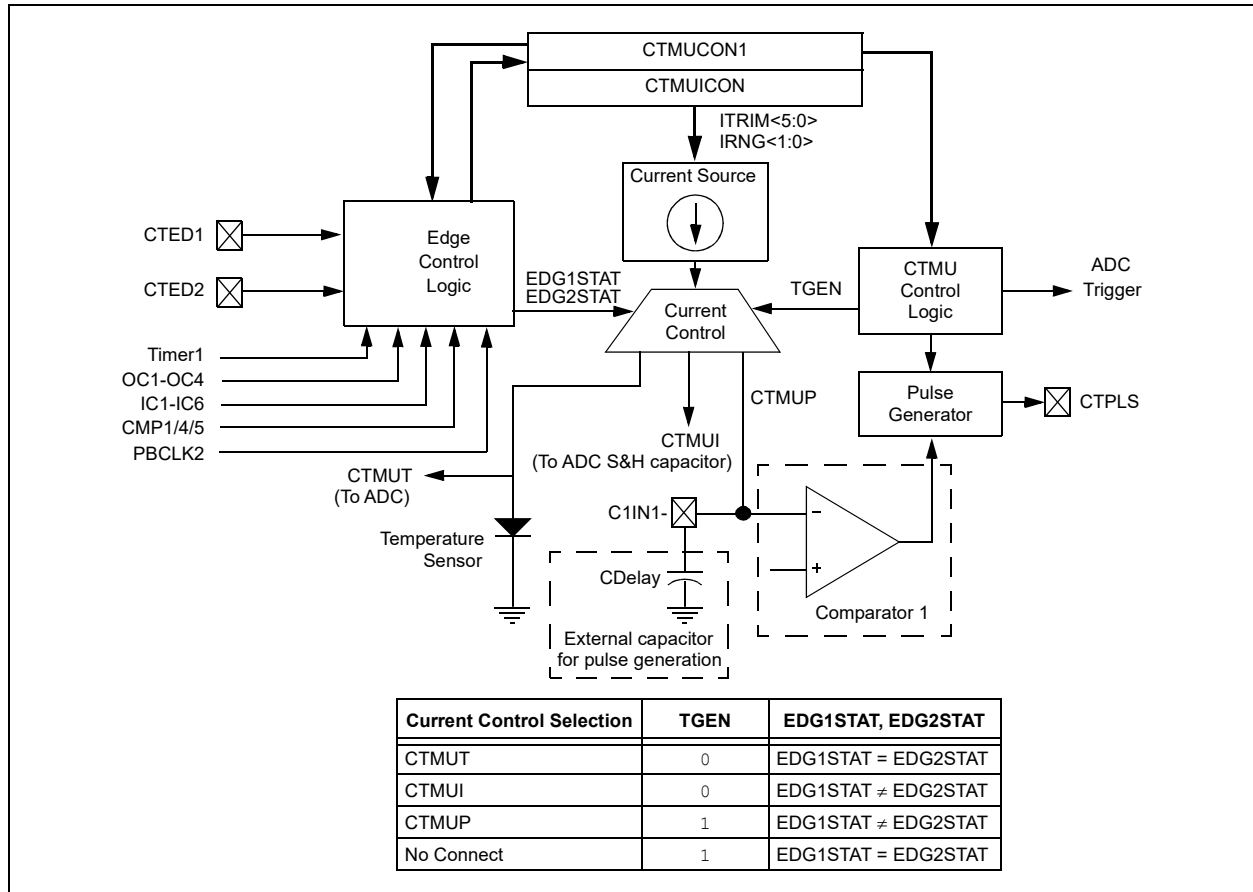
or generate output pulses with a specific time delay. The CTMU is ideal for interfacing with capacitive-based sensors.

The CTMU module includes the following key features:

- Two channels are available for capacitive or time measurement input
- On-chip precision current source
- 16-edge input trigger sources
- Selection of edge or level-sensitive inputs
- Polarity control for each edge source
- Control of edge sequence
- Control of response to edges
- High precision time measurement
- Time delay of external or internal signal asynchronous to system clock
- Integrated temperature sensing diode
- Control of current source during auto-sampling
- Four current source ranges
- Time measurement resolution of one nanosecond
- Up to 39 inputs for capacitive measurement

A block diagram of the CTMU is shown in [Figure 26-1](#).

**FIGURE 26-1: CTMU BLOCK DIAGRAM**





26.1 Control Registers

TABLE 26-1: CTMU REGISTER MAP

| Virtual Address<br>(BF82_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits    |         |              |       |       |          |          |          |            |         |              |      |      |           | All Resets |      |
|-----------------------------|---------------------------------|-----------|---------|---------|--------------|-------|-------|----------|----------|----------|------------|---------|--------------|------|------|-----------|------------|------|
|                             |                                 |           | 31/15   | 30/14   | 29/13        | 28/12 | 27/11 | 26/10    | 25/9     | 24/8     | 23/7       | 22/6    | 21/5         | 20/4 | 19/3 | 18/2      |            | 17/1 |
| D000                        | CTMUCON                         | 31:16     | EDG1MOD | EDG1POL | EDG1SEL<3:0> |       |       |          | EDG2STAT | EDG1STAT | EDG2MOD    | EDG2POL | EDG2SEL<3:0> |      |      | —         | —          | 0000 |
|                             |                                 | 15:0      | ON      | —       | SIDLE        | TGEN  | EDGEN | EDGSEQEN | IDISSEN  | CTRIG    | ITRIM<5:0> |         |              |      |      | IRNG<1:0> |            | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 26-1: CTMUCON: CTMU CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4      | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1          | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|---------------------|----------------|----------------|------------------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0          | R/W-0          | R/W-0                  | R/W-0         |
|           | EDG1MOD        | EDG1POL        | EDG1SEL<3:0>   |                     |                |                | EDG2STAT               | EDG1STAT      |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0          | R/W-0          | U-0                    | U-0           |
|           | EDG2MOD        | EDG2POL        | EDG2SEL<3:0>   |                     |                |                | —                      | —             |
| 15:8      | R/W-0          | U-0            | R/W-0          | R/W-0               | R/W-0          | R/W-0          | R/W-0                  | R/W-0         |
|           | ON             | —              | SIDLE          | TGEN <sup>(1)</sup> | EDGEN          | EDGSEQEN       | IDISSEN <sup>(2)</sup> | CTTRIG        |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0               | R/W-0          | R/W-0          | R/W-0                  | R/W-0         |
|           | ITRIM<5:0>     |                |                |                     |                |                | IRNG<1:0>              |               |

**Legend:**

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31 **EDG1MOD:** Edge 1 Edge Sampling Select bit

1 = Reserved  
0 = Input is level-sensitive

bit 30 **EDG1POL:** Edge 1 Polarity Select bit

1 = Edge 1 programmed for a logic high level response  
0 = Edge 1 programmed for a logic low level response

bit 29-26 **EDG1SEL<3:0>:** Edge 1 Source Select bits

1111 = C5OUT Capture Event is selected  
1110 = C4OUT pin is selected  
1101 = C1OUT pin is selected  
1100 = PBCLK2 is selected  
1011 = IC5 Capture Event is selected  
1010 = IC4 Capture Event is selected  
1001 = IC3 pin is selected  
1000 = IC2 pin is selected  
0111 = IC1 pin is selected  
0110 = OC4 pin is selected  
0101 = OC3 pin is selected  
0100 = OC2 pin is selected  
0011 = CTED1 pin is selected  
0010 = CTED2 pin is selected  
0001 = OC1 Compare Event is selected  
0000 = Timer1 Event is selected

**Note 1:** When this bit is set for Pulse Delay Generation, the EDG2SEL<3:0> bits must be set to '1101' to select C1OUT.

**2:** The ADC module Sample and Hold capacitor is not automatically discharged between sample/conversion cycles. Software using the ADC as part of a capacitive measurement, must discharge the ADC capacitor before conducting the measurement. The IDISSEN bit, when set to '1', performs this function. The ADC module must be sampling while the IDISSEN bit is active to connect the discharge sink to the capacitor array.

**3:** Refer to the CTMU Current Source Specifications (Table 36-43) in 36.0 "Electrical Characteristics" for current values.

**4:** This bit setting is not available for the CTMU temperature diode.

**5:** For CTMU temperature measurements on this range, ADC sampling time ≥ 1.6 μs.

**6:** For CTMU temperature measurements on this range, ADC sampling time ≥ 300 ns.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 26-1: CTMUCON: CTMU CONTROL REGISTER (CONTINUED)

- bit 25 **EDG2STAT:** Edge 2 Status bit  
Indicates the status of Edge 2 and can be written to control edge source  
1 = Edge 2 event has occurred  
0 = Edge 2 event has not occurred
- bit 24 **EDG1STAT:** Edge 1 Status bit  
Indicates the status of Edge 1 and can be written to control edge source  
1 = Edge 1 event has occurred  
0 = Edge 1 event has not occurred
- bit 23 **EDG2MOD:** Edge 2 Edge Sampling Select bit  
1 = Reserved  
0 = Input is level-sensitive
- bit 22 **EDG2POL:** Edge 2 Polarity Select bit  
1 = Edge 2 programmed for a high level response  
0 = Edge 2 programmed for a low level response
- bit 21-18 **EDG2SEL<3:0>:** Edge 2 Source Select bits  
1111 = C5OUT Capture Event is selected  
1110 = C4OUT pin is selected  
1101 = C1OUT pin is selected  
1100 = IC6 Capture Event is selected  
1011 = IC5 Capture Event is selected  
1010 = IC4 Capture Event is selected  
1001 = IC3 pin is selected  
1000 = IC2 pin is selected  
0111 = IC1 pin is selected  
0110 = OC4 pin is selected  
0101 = OC3 pin is selected  
0100 = OC2 pin is selected  
0011 = CTED1 pin is selected  
0010 = CTED2 pin is selected  
0001 = OC1 Compare Event is selected  
0000 = Timer1 Event is selected
- bit 17-16 **Unimplemented:** Read as '0'
- bit 15 **ON:** ON Enable bit  
1 = Module is enabled  
0 = Module is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **SIDLE:** Stop-in-Idle-Mode  
1 = Discontinue Module Operation when device enters idle mode.  
0 = Continue module operation in idle mode.

- Note 1:** When this bit is set for Pulse Delay Generation, the EDG2SEL<3:0> bits must be set to '1101' to select C1OUT.
- 2:** The ADC module Sample and Hold capacitor is not automatically discharged between sample/conversion cycles. Software using the ADC as part of a capacitive measurement, must discharge the ADC capacitor before conducting the measurement. The IDISSEN bit, when set to '1', performs this function. The ADC module must be sampling while the IDISSEN bit is active to connect the discharge sink to the capacitor array.
- 3:** Refer to the CTMU Current Source Specifications ([Table 36-43](#)) in **36.0 "Electrical Characteristics"** for current values.
- 4:** This bit setting is not available for the CTMU temperature diode.
- 5:** For CTMU temperature measurements on this range, ADC sampling time  $\geq 1.6 \mu\text{s}$ .
- 6:** For CTMU temperature measurements on this range, ADC sampling time  $\geq 300 \text{ ns}$ .

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 26-1: CTMUCON: CTMU CONTROL REGISTER (CONTINUED)

- bit 12 **TGEN:** Time Generation Enable bit<sup>(1)</sup>  
1 = Enables edge delay generation  
0 = Disables edge delay generation
- bit 11 **EDGEN:** Edge Enable bit  
1 = Edges are not blocked  
0 = Edges are blocked
- bit 10 **EDGSEQEN:** Edge Sequence Enable bit  
1 = Edge 1 must occur before Edge 2 can occur  
0 = No edge sequence is needed
- bit 9 **IDISSEN:** Analog Current Source Control bit<sup>(2)</sup>  
1 = Analog current source output is grounded  
0 = Analog current source output is not grounded
- bit 8 **CTTRIG:** Trigger Control bit  
1 = Trigger output is enabled  
0 = Trigger output is disabled
- bit 7-2 **ITRIM<5:0>:** Current Source Trim bits  
011111 = Maximum positive change from nominal current  
011110  
.  
.  
.  
000001 = Minimum positive change from nominal current  
000000 = Nominal current output specified by IRNG<1:0>  
111111 = Minimum negative change from nominal current  
.  
.  
.  
100010  
100001 = Maximum negative change from nominal current
- bit 1-0 **IRNG<1:0>:** Current Range Select bits<sup>(3)</sup>  
11 = 100 times base current (i.e., 0.55  $\mu$ A Typical)<sup>(6)</sup>  
10 = 10 times base current (i.e., 5.5  $\mu$ A Typical)<sup>(5)</sup>  
01 = Base current level (i.e., 0.55  $\mu$ A Typical)<sup>(4)</sup>  
00 = 1000 times base current (i.e., 550  $\mu$ A Typical)<sup>(4)</sup>

- Note 1:** When this bit is set for Pulse Delay Generation, the EDG2SEL<3:0> bits must be set to '1101' to select C1OUT.
- 2:** The ADC module Sample and Hold capacitor is not automatically discharged between sample/conversion cycles. Software using the ADC as part of a capacitive measurement, must discharge the ADC capacitor before conducting the measurement. The IDISSEN bit, when set to '1', performs this function. The ADC module must be sampling while the IDISSEN bit is active to connect the discharge sink to the capacitor array.
- 3:** Refer to the CTMU Current Source Specifications (Table 36-43) in **36.0 "Electrical Characteristics"** for current values.
- 4:** This bit setting is not available for the CTMU temperature diode.
- 5:** For CTMU temperature measurements on this range, ADC sampling time  $\geq 1.6 \mu$ s.
- 6:** For CTMU temperature measurements on this range, ADC sampling time  $\geq 300$  ns.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 27.0 CONTROL DIGITAL-TO-ANALOG CONVERTER (CDAC)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 45. “Control Digital-to-Analog Converter (CDAC)”** (DS60001327), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The PIC32MK GPG/MCJ with CAN FD Family Control Digital-to-Analog Converter (CDAC) generates analog voltage corresponding to the digital inputs. The voltage can be used as a reference source for comparators or can be used as an offset to an op amp. This module is targeted for control applications, as opposed to other DAC modules, which are used for audio applications.

The following are key features of the CDAC module:

- Wide voltage range (1.8V to 3.6V)
- 12-bit resolution
- Fast conversion times, 1 Msp/s
- Buffered output for comparator use

**Note:** For additional information on conversion time, sampling rate, module turn-on time and glitch reduction circuit characteristics, refer to **36.0 “Electrical Characteristics”**.

Figure 27-1 illustrates the functional block diagram of the CDAC module.

**FIGURE 27-1: CDAC BLOCK DIAGRAM**

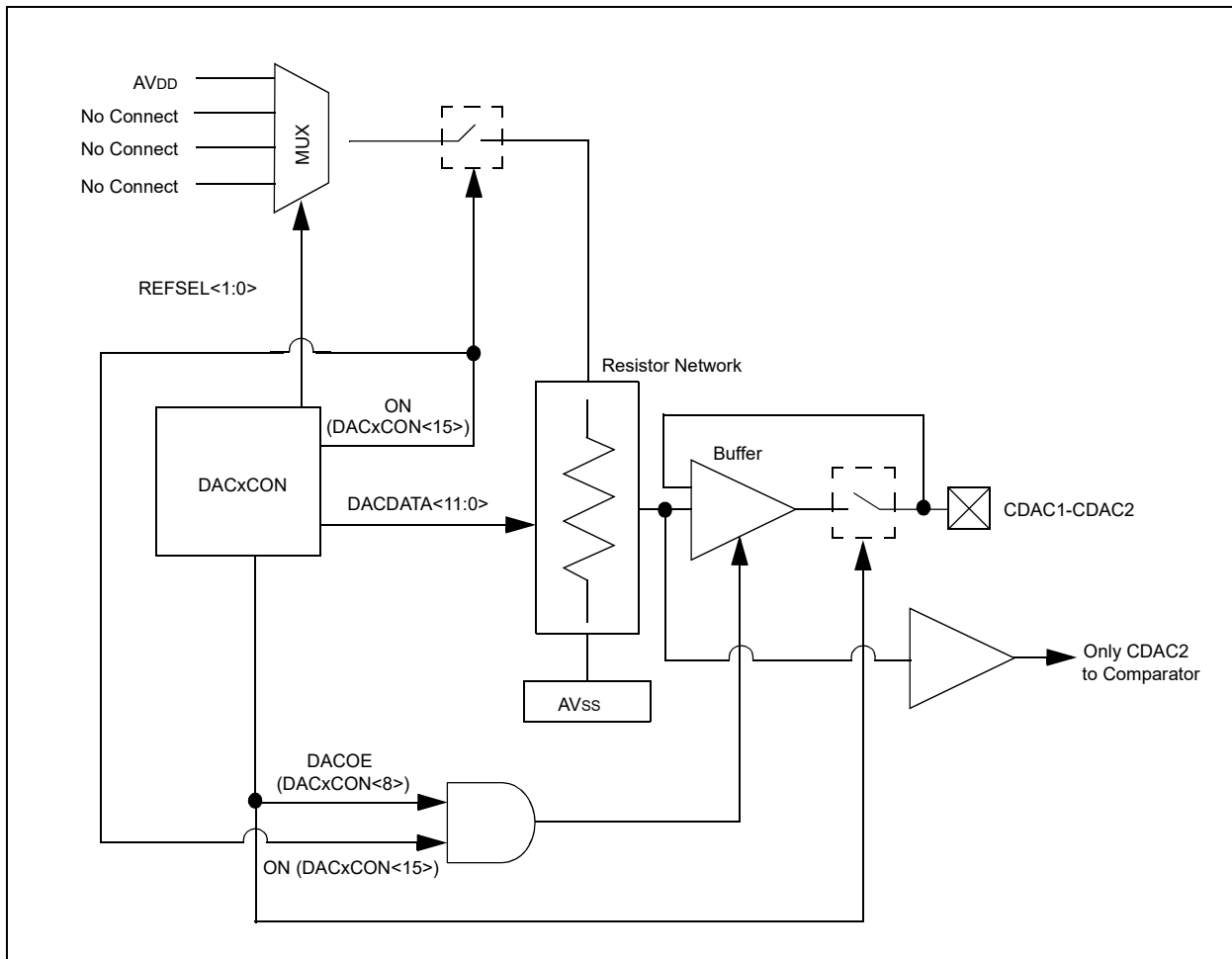


TABLE 27-1: CDAC REGISTER MAP

| Virtual Address | Register Name <sup>(1)</sup> | Bit Range | Bits  |       |       |       |              |       |      |       |      |      |      |      |      |      |      | All Resets  |
|-----------------|------------------------------|-----------|-------|-------|-------|-------|--------------|-------|------|-------|------|------|------|------|------|------|------|-------------|
|                 |                              |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11        | 26/10 | 25/9 | 24/8  | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |             |
| BF84<br>C400    | DAC1CON                      | 31:16     | —     | —     | —     | —     | DACDAT<11:0> |       |      |       |      |      |      |      |      |      |      | 0000        |
|                 |                              | 15:0      | ON    | —     | —     | —     | —            | —     | —    | DACOE | —    | —    | —    | —    | —    | —    | —    | REFSEL<1:0> |
| BF84<br>C600    | DAC2CON                      | 31:16     | —     | —     | —     | —     | DACDAT<11:0> |       |      |       |      |      |      |      |      |      |      | 0000        |
|                 |                              | 15:0      | ON    | —     | —     | —     | —            | —     | —    | DACOE | —    | —    | —    | —    | —    | —    | —    | REFSEL<1:0> |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 27-1: DAC<sub>x</sub>CON: CDAC CONTROL REGISTER 'x' ('x' = 2 THROUGH 3)**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3              | Bit 26/18/10/2 | Bit 25/17/9/1                | Bit 24/16/8/0        |
|-----------|----------------------------|----------------|----------------|----------------|-----------------------------|----------------|------------------------------|----------------------|
| 31:24     | U-0                        | U-0            | U-0            | U-0            | R/W-0                       | R/W-0          | R/W-0                        | R/W-0                |
|           | —                          | —              | —              | —              | DACDAT<11:8> <sup>(1)</sup> |                |                              |                      |
| 23:16     | R/W-0                      | R/W-0          | R/W-0          | R/W-0          | R/W-0                       | R/W-0          | R/W-0                        | R/W-0                |
|           | DACDAT<7:0> <sup>(1)</sup> |                |                |                |                             |                |                              |                      |
| 15:8      | R/W-0                      | U-0            | U-0            | U-0            | U-0                         | U-0            | U-0                          | R/W-0                |
|           | ON <sup>(1)</sup>          | —              | —              | —              | —                           | —              | —                            | DACOE <sup>(1)</sup> |
| 7:0       | U-0                        | U-0            | U-0            | U-0            | U-0                         | U-0            | R/W-0                        | R/W-0                |
|           | —                          | —              | —              | —              | —                           | —              | REFSEL<1:0> <sup>(1,2)</sup> |                      |

|                   |  |
|-------------------|--|
| <b>Legend:</b>    | y = Value set from Configuration bits on POR                       |
| R = Readable bit  | W = Writable bit      U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown |

bit 31-28 **Unimplemented:** Read as '0'

bit 27-16 **DACDAT<11:0>:** CDAC Data Port bits<sup>(1)</sup>  
Data input register bits for the CDAC.

bit 15 **ON:** CDAC Enable bit  
1 = The CDAC is enabled  
0 = The CDAC is disabled

bit 14-9 **Unimplemented:** Read as '0'

bit 8 **DACOE:** CDAC Output Buffer Enable bit  
1 = Output is enabled; CDAC voltage is connected to the pin  
0 = Output is disabled; drive to pin is floating

bit 7-2 **Unimplemented:** Read as '0'

bit 1-0 **REFSEL<1:0>:** Reference Source Select bits<sup>(1,2)</sup>  
11 = Positive reference voltage = AVDD  
10 = No reference selected (no reference current consumption)  
01 = No reference selected (no reference current consumption)  
00 = No reference selected (no reference current consumption)

**Note 1:** To minimize CDAC start-up output transients, configure the DACDATA<15:0>, DACOE, and REFSEL<1:0> bits prior to enabling the CDAC (prior to making DACON = 1). Also, remember to wait TON time, after enabling the CDAC. This time is required to allow the CDAC output to stabilize. Refer to [36.0 "Electrical Characteristics"](#) for the TON specification.

**2:** If the ON bit is '0', the reference source is disconnected from the internal resistor network.



## 28.0 QUADRATURE ENCODER INTERFACE (QEI)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 43. “Quadrature Encoder Interface (QEI)”** (DS60001346), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

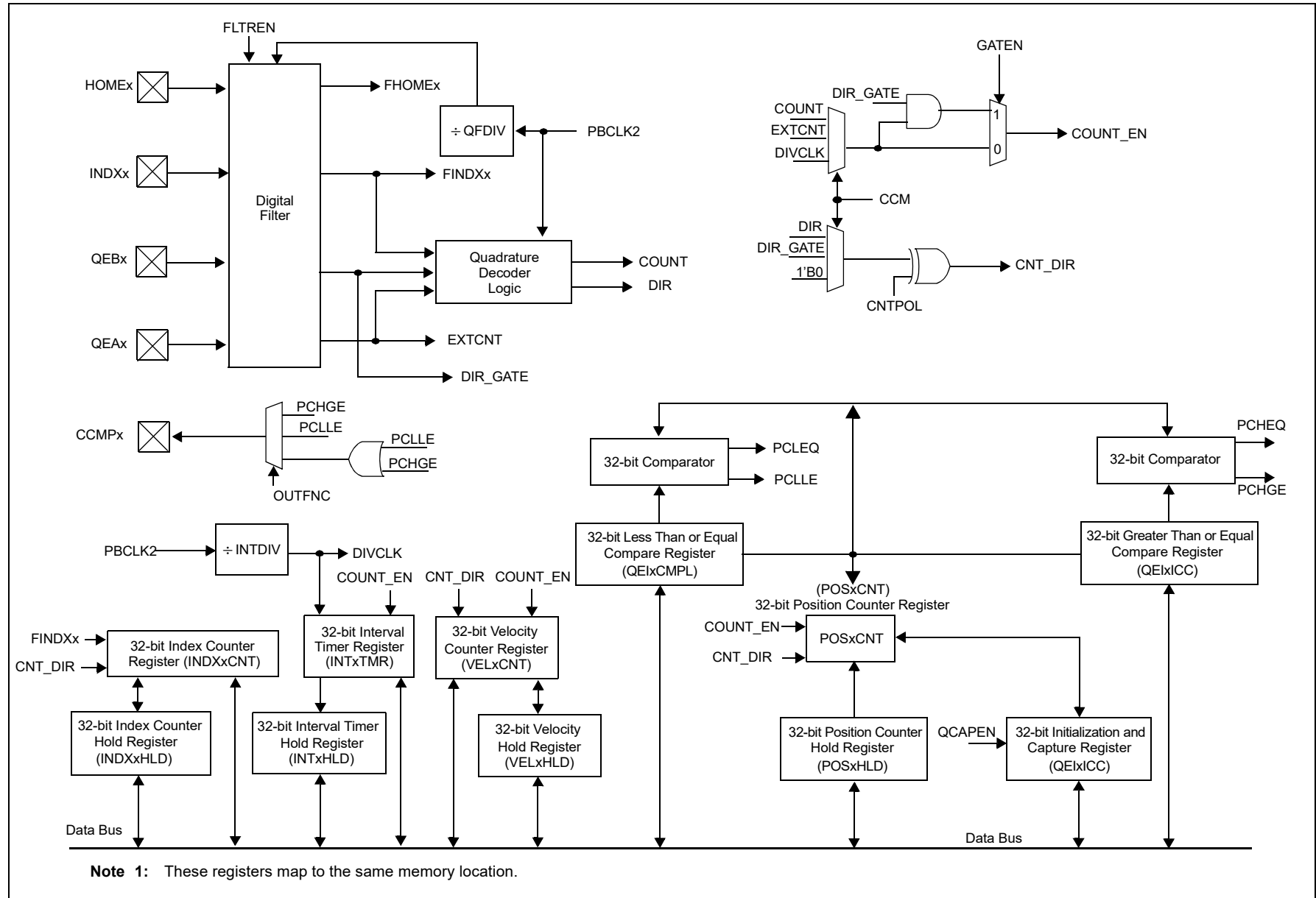
This chapter describes the Quadrature Encoder Interface (QEI) module and associated operational modes. The QEI module provides the interface to incremental encoders for obtaining mechanical position data.

The QEI module consists of the following major features:

- Four input pins: two phase signals, an index pulse and a home pulse
- Programmable digital noise filters on inputs
- Quadrature decoder providing counter pulses and count direction
- Count direction status
- 4x count resolution
- Index (INDX) pulse to reset the position counter
- General purpose 32-bit Timer/Counter mode
- Interrupts generated by QEI or counter events
- 32-bit velocity counter
- 32-bit position counter
- 32-bit index pulse counter
- 32-bit interval timer
- 32-bit position Initialization/Capture register
- 32-bit Compare Less Than and Greater Than registers
- External Up/Down Count mode
- External Gated Count mode
- External Gated Timer mode
- Interval Timer mode

Figure 28-1 illustrates the QEI block diagram.

**FIGURE 28-1: QEI BLOCK DIAGRAM**



## 28.1 QEI Control Registers

**TABLE 28-1: QE1 THROUGH QE6 REGISTER MAP**

| Virtual Address (BF82_#) | Register Name(1) | Bit Range | Bits           |        |            |            |          |             |          |          |        |        |             |          |        |        |        | All Resets |        |      |
|--------------------------|------------------|-----------|----------------|--------|------------|------------|----------|-------------|----------|----------|--------|--------|-------------|----------|--------|--------|--------|------------|--------|------|
|                          |                  |           | 31/15          | 30/14  | 29/13      | 28/12      | 27/11    | 26/10       | 25/9     | 24/8     | 23/7   | 22/6   | 21/5        | 20/4     | 19/3   | 18/2   | 17/1   |            | 16/0   |      |
| B200                     | QE1CON           | 31:16     | —              | —      | —          | —          | —        | —           | —        | —        | —      | —      | —           | —        | —      | —      | —      | —          | 0000   |      |
|                          |                  | 15:0      | QEIEN          | —      | QEISIDL    | PIMOD<2:0> |          |             | IMV<1:0> |          |        | —      | INTDIV<2:0> |          |        | CNTPOL | GATEN  | CCM<1:0>   |        | 0000 |
| B210                     | QE1IOC           | 31:16     | —              | —      | —          | —          | —        | —           | —        | —        | —      | —      | —           | —        | —      | —      | —      | —          | HCAPEN | 0000 |
|                          |                  | 15:0      | QCAPEN         | FLTREN | QFDIV<2:0> |            |          | OUTFNC<1:0> |          |          | SWPAB  | HOMPOL | IDXPOL      | QEBPOL   | QEAPOL | HOME   | INDEX  | QEB        | QEA    | 0000 |
| B220                     | QE1STAT          | 31:16     | —              | —      | —          | —          | —        | —           | —        | —        | —      | —      | —           | —        | —      | —      | —      | —          | —      | 0000 |
|                          |                  | 15:0      | —              | —      | PCHEQIRQ   | PCHEQIEN   | PCLEQIRQ | PCLEQIEN    | POSOVIRQ | POSOVIEN | PCIIRQ | PCIIEN | VELOVIRQ    | VELOVIEN | HOMIRQ | HOMIEN | IDXIRQ | IDXIEN     | —      | 0000 |
| B230                     | POS1CNT          | 31:16     | POSCNT<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | POSCNT<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B240                     | POS1HLD          | 31:16     | POSHLD<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | POSHLD<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B250                     | VEL1CNT          | 31:16     | VELCNT<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | VELCNT<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B260                     | VEL1HLD          | 31:16     | VELHLD<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | VELHLD<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B270                     | INT1TMR          | 31:16     | INTTMR<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | INTTMR<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B280                     | INT1HLD          | 31:16     | INTHLD<31:16>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | INTHLD<15:0>   |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B290                     | INDX1CNT         | 31:16     | INDXCNT<31:16> |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | INDXCNT<15:0>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B2A0                     | INDX1HLD         | 31:16     | INDXHLD<31:16> |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | INDXHLD<15:0>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B2B0                     | QE1ICC           | 31:16     | QEIIICC<31:16> |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | QEIIICC<15:0>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B2C0                     | QE1CMPL          | 31:16     | QEICMPL<31:16> |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
|                          |                  | 15:0      | QEICMPL<15:0>  |        |            |            |          |             |          |          |        |        |             |          |        |        |        | 0000       |        |      |
| B400                     | QE2CON           | 31:16     | —              | —      | —          | —          | —        | —           | —        | —        | —      | —      | —           | —        | —      | —      | —      | —          | —      | 0000 |
|                          |                  | 15:0      | QEIEN          | —      | QEISIDL    | PIMOD<2:0> |          |             | IMV<1:0> |          |        | —      | INTDIV<2:0> |          |        | CNTPOL | GATEN  | CCM<1:0>   |        | 0000 |
| B410                     | QE2IOC           | 31:16     | —              | —      | —          | —          | —        | —           | —        | —        | —      | —      | —           | —        | —      | —      | —      | —          | HCAPEN | 0000 |
|                          |                  | 15:0      | QCAPEN         | FLTREN | QFDIV<2:0> |            |          | OUTFNC<1:0> |          |          | SWPAB  | HOMPOL | IDXPOL      | QEBPOL   | QEAPOL | HOME   | INDEX  | QEB        | QEA    | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

TABLE 28-1: QE1 THROUGH QE16 REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register<br>Name(1) | Bit Range | Bits            |        |            |            |          |             |          |          |        |             |          |          |        |        |          | All Resets |      |
|-----------------------------|---------------------|-----------|-----------------|--------|------------|------------|----------|-------------|----------|----------|--------|-------------|----------|----------|--------|--------|----------|------------|------|
|                             |                     |           | 31/15           | 30/14  | 29/13      | 28/12      | 27/11    | 26/10       | 25/9     | 24/8     | 23/7   | 22/6        | 21/5     | 20/4     | 19/3   | 18/2   | 17/1     |            | 16/0 |
| B420                        | QE12STAT            | 31:16     | —               | —      | —          | —          | —        | —           | —        | —        | —      | —           | —        | —        | —      | —      | —        | —          | 0000 |
|                             |                     | 15:0      | —               | —      | PCHEQIRQ   | PCHEQIEN   | PCLEQIRQ | PCLEQIEN    | POSOVIRQ | POSOVIEN | PCIIRQ | PCIIEN      | VELOVIRQ | VELOVIEN | HOMIRQ | HOMIEN | IDXIRQ   | IDXIEN     | 0000 |
| B430                        | POS2CNT             | 31:16     | POSCNT<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | POSCNT<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B440                        | POS2HLD             | 31:16     | POSHLD<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | POSHLD<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B450                        | VEL2CNT             | 31:16     | VELCNT<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | VELCNT<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B460                        | VEL2HLD             | 31:16     | VELHLD<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | VELHLD<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B470                        | INT2TMR             | 31:16     | INTTMR<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | INTTMR<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B480                        | INT2HLD             | 31:16     | INTHLD<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | INTHLD<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B490                        | INDX2CNT            | 31:16     | INDXCNT<31:16>  |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | INDXCNT<15:0>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B4A0                        | INDX2HLD            | 31:16     | INDXHLD<31:16>  |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | INDXHLD<15:0>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B4B0                        | QE12ICC             | 31:16     | QE12ICC<31:16>  |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | QE12ICC<15:0>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B4C0                        | QE12Cmpl            | 31:16     | QE12Cmpl<31:16> |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | QE12Cmpl<15:0>  |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B600                        | QE13CON             | 31:16     | —               | —      | —          | —          | —        | —           | —        | —        | —      | —           | —        | —        | —      | —      | —        | —          | 0000 |
|                             |                     | 15:0      | QE13EN          | —      | QE13SIDL   | PIMOD<2:0> |          |             | IMV<1:0> |          | —      | INTDIV<2:0> |          |          | CNTPOL | GATEN  | CCM<1:0> |            | 0000 |
| B610                        | QE13IOC             | 31:16     | —               | —      | —          | —          | —        | —           | —        | —        | —      | —           | —        | —        | —      | —      | —        | —          | 0000 |
|                             |                     | 15:0      | QCAPEN          | FLTREN | QFDIV<2:0> |            |          | OUTFNC<1:0> |          | SWPAB    | HOMPOL | IDXPOL      | QEBPOL   | QEAPOL   | HOME   | INDEX  | QEB      | QEA        | 0000 |
| B620                        | QE13STAT            | 31:16     | —               | —      | —          | —          | —        | —           | —        | —        | —      | —           | —        | —        | —      | —      | —        | —          | 0000 |
|                             |                     | 15:0      | —               | —      | PCHEQIRQ   | PCHEQIEN   | PCLEQIRQ | PCLEQIEN    | POSOVIRQ | POSOVIEN | PCIIRQ | PCIIEN      | VELOVIRQ | VELOVIEN | HOMIRQ | HOMIEN | IDXIRQ   | IDXIEN     | 0000 |
| B630                        | POS3CNT             | 31:16     | POSCNT<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | POSCNT<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
| B640                        | POS3HLD             | 31:16     | POSHLD<31:16>   |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |
|                             |                     | 15:0      | POSHLD<15:0>    |        |            |            |          |             |          |          |        |             |          |          |        |        |          | 0000       |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See 10.2 "CLR, SET, and INV Registers" for more information.

**TABLE 28-1: QE1 THROUGH QE16 REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF82_#) | Register<br>Name(1) | Bit Range | Bits           |       |       |       |       |       |      |      |      |      |      |      |      | All Resets |
|-----------------------------|---------------------|-----------|----------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------------|
|                             |                     |           | 31/15          | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 |            |
| B650                        | VEL3CNT             | 31:16     | VELCNT<31:16>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | VELCNT<15:0>   |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B660                        | VEL3HLD             | 31:16     | VELHLD<31:16>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | VELHLD<15:0>   |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B670                        | INT3TMR             | 31:16     | INTTMR<31:16>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | INTTMR<15:0>   |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B680                        | INT3HLD             | 31:16     | INTHLD<31:16>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | INTHLD<15:0>   |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B690                        | INDX3CNT            | 31:16     | INDXCNT<31:16> |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | INDXCNT<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B6A0                        | INDX3HLD            | 31:16     | INDXHLD<31:16> |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | INDXHLD<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B6B0                        | QEI3ICC             | 31:16     | QEIIICC<31:16> |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | QEIIICC<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
| B6C0                        | QEI3CMPL            | 31:16     | QEICMPL<31:16> |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |
|                             |                     | 15:0      | QEICMPL<15:0>  |       |       |       |       |       |      |      |      |      |      |      |      | 0000       |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** All registers in this table have corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 28-1: QEIXCON: QEIX CONTROL REGISTER ('X' = 1-3) (CONTINUED)

- bit 6-4 **INTDIV<2:0>**: Timer Input Clock Prescale Select bits (Interval timer, Main timer (position counter), velocity counter and index counter internal clock divider select)<sup>(3)</sup>
- 111 = 1:128 prescale value
  - 110 = 1:64 prescale value
  - 101 = 1:32 prescale value
  - 100 = 1:16 prescale value
  - 011 = 1:8 prescale value
  - 010 = 1:4 prescale value
  - 001 = 1:2 prescale value
  - 000 = 1:1 prescale value
- bit **CNTPOL**: Position and Index Counter/Timer Direction Select bit
- 1 = Counter direction is negative unless modified by external Up/Down signal
  - 0 = Counter direction is positive unless modified by external Up/Down signal
- bit **GATEN**: External Count Gate Enable bit
- 1 = External gate signal controls position counter operation
  - 0 = External gate signal does not affect position counter/timer operation
- bit **CCM<1:0>**: Counter Control Mode Selection bits
- 11 = Internal Timer mode with optional QEB external clock gating input control based on GATEN. QEB High = Timer Run, QEB Low = Timer Stop.
  - 10 = QEA is the external clock input, QEB is optional clock gating input control based on GATEN. QEB High = Clock Run, QEB Low = Clock Stop.
  - 01 = QEA is the external clock input, QEB is external UP/DN direction input. (QEB High = Count Up, QEB Low = Count Down)
  - 00 = Quadrature Encoder Interface Count mode (x4 mode)

- Note 1:** When CCM equals modes '01', '10', and '11', all of the QEI counters operate as timers and the PIMOD<2:0> bits are ignored.
- 2:** When CCM = 00 and QEA and QEB values match Index Match Value (IMV), the POSxCNTH and POSxCNTL registers are reset.
- 3:** The selected clock rate should be at least twice the expected maximum quadrature count rate.





# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 28-2: QEIXIOC: QEIX I/O CONTROL REGISTER ('X' = 1-3) (CONTINUED)

- bit 4     **QEAPOL:** QEAX Input Polarity Select bit  
          1 = Input is inverted  
          0 = Input is not inverted
- bit 3     **HOME:** Status of HOME<sub>x</sub> Input Pin after Polarity Control bit (read-only)  
          1 = Pin is at logic '1', if HOMPOL bit is set to '0'  
              Pin is at logic '0', if HOMPOL bit is set to '1'  
          0 = Pin is at logic '0', if HOMPOL bit is set to '0'  
              Pin is at logic '1', if HOMPOL bit is set to '1'
- bit 2     **INDEX:** Status of IND<sub>X</sub> Input Pin after Polarity Control bit (Read-Only)  
          1 = Pin is at logic '1', if IDXPOL bit is set to '0'  
              Pin is at logic '0', if IDXPOL bit is set to '1'  
          0 = Pin is at logic '0', if IDXPOL bit is set to '0'  
              Pin is at logic '1', if IDXPOL bit is set to '1'
- bit 1     **QEB:** Status of QEB<sub>x</sub> Input Pin after Polarity Control and SWPAB Pin Swapping bit (read-only)  
          1 = Physical pin QEB is at logic '1', if QEBPOL bit is set to '0' and SWPAB bit is set to '0'  
              Physical pin QEB is at logic '0', if QEBPOL bit is set to '1' and SWPAB bit is set to '0'  
              Physical pin QEA is at logic '1', if QEBPOL bit is set to '0' and SWPAB bit is set to '1'  
              Physical pin QEA is at logic '0', if QEBPOL bit is set to '1' and SWPAB bit is set to '1'  
  
          0 = Physical pin QEB is at logic '0', if QEBPOL bit is set to '0' and SWPAB bit is set to '0'  
              Physical pin QEB is at logic '1', if QEBPOL bit is set to '1' and SWPAB bit is set to '0'  
              Physical pin QEA is at logic '0', if QEBPOL bit is set to '0' and SWPAB bit is set to '1'  
              Physical pin QEA is at logic '1', if QEBPOL bit is set to '1' and SWPAB bit is set to '1'
- bit 0     **QEA:** Status of QEA<sub>x</sub> Input Pin after Polarity Control and SWPAB Pin Swapping bit (read-only)  
          1 = Physical pin QEA is at logic '1', if QEAPOL bit is set to '0' and SWPAB bit is set to '0'  
              Physical pin QEA is at logic '0', if QEAPOL bit is set to '1' and SWPAB bit is set to '0'  
              Physical pin QEB is at logic '1', if QEAPOL bit is set to '0' and SWPAB bit is set to '1'  
              Physical pin QEB is at logic '0', if QEAPOL bit is set to '1' and SWPAB bit is set to '1'  
  
          0 = Physical pin QEA is at logic '0', if QEAPOL bit is set to '0' and SWPAB bit is set to '0'  
              Physical pin QEA is at logic '1', if QEAPOL bit is set to '1' and SWPAB bit is set to '0'  
              Physical pin QEB is at logic '0', if QEAPOL bit is set to '0' and SWPAB bit is set to '1'  
              Physical pin QEB is at logic '1', if QEAPOL bit is set to '1' and SWPAB bit is set to '1'

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 28-3: QEIXSTAT: QEIX STATUS REGISTER ('X' = 1-3)**

| Bit Range | Bit 31/23/15/7        | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                   | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                     | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0                   | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                     | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | U-0                   | U-0            | RC-0, HS       | R/W-0          | RC-0, HS       | R/W-0          | RC-0, HS      | R/W-0         |
|           | —                     | —              | PCHEQIRQ       | PCHEQIEN       | PCLEQIRQ       | PCLEQIEN       | POSOVIRQ      | POSOVIEN      |
| 7:0       | RC-0, HS              | R/W-0          | RC-0, HS       | R/W-0          | RC-0, HS       | R/W-0          | RC-0, HS      | R/W-0         |
|           | PCIIRQ <sup>(1)</sup> | PCIEN          | VELOVIRQ       | VELOVIEN       | HOMIRQ         | HOMIEN         | IDXIRQ        | IDXIEN        |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-14 **Unimplemented:** Read as '0'

bit 13 **PCHEQIRQ:** Position Counter Greater Than or Equal Compare Status bit

1 = POSxCNT > QEIXICCH  
 0 = POSxCNT ≤ QEIXICCH

bit 12 **PCHEQIEN:** Position Counter Greater Than or Equal Compare Interrupt Enable bit

1 = Interrupt is enabled  
 0 = Interrupt is disabled

bit 11 **PCLEQIRQ:** Position Counter Less Than or Equal Compare Status bit

1 = POSxCNT < QEIXCMPL  
 0 = POSxCNT ≥ QEIXCMPL

bit 10 **PCLEQIEN:** Position Counter Less Than or Equal Compare Interrupt Enable bit

1 = Interrupt is enabled  
 0 = Interrupt is disabled

bit 9 **POSOVIRQ:** Position Counter Overflow Status bit

1 = Overflow has occurred  
 0 = No overflow has occurred

bit 8 **POSOVIEN:** Position Counter Overflow Interrupt Enable bit

1 = Interrupt is enabled  
 0 = Interrupt is disabled

bit 7 **PCIIRQ:** Position Counter (Homing) Initialization Process Complete Status bit<sup>(1)</sup>

1 = POSxCNT was reinitialized  
 0 = POSxCNT was not reinitialized

bit 6 **PCIEN:** Position Counter (Homing) Initialization Process Complete Interrupt Enable bit

1 = Interrupt is enabled  
 0 = Interrupt is disabled

bit 5 **VELOVIRQ:** Velocity Counter Overflow Status bit

1 = Overflow has occurred  
 0 = No overflow has not occurred

bit 4 **VELOVIEN:** Velocity Counter Overflow Interrupt Enable bit

1 = Interrupt is enabled  
 0 = Interrupt is disabled

**Note 1:** This status bit in only applies to PIMOD<2:0> modes '011' and '100'.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 28-3: QEIXSTAT: QEIX STATUS REGISTER ('X' = 1-3) (CONTINUED)

- bit 3     **HOMIRQ**: Status Flag for Home Event Status bit  
          1 = Home event has occurred  
          0 = No Home event has occurred
- bit 2     **HOMIEN**: Home Input Event Interrupt Enable bit  
          1 = Interrupt is enabled  
          0 = Interrupt is disabled
- bit 1     **IDXIRQ**: Status Flag for Index Event Status bit  
          1 = Index event has occurred  
          0 = No Index event has occurred
- bit 0     **IDXIEN**: Index Input Event Interrupt Enable bit  
          1 = Interrupt is enabled  
          0 = Interrupt is disabled

**Note 1:** This status bit in only applies to PIMOD<2:0> modes '011' and '100'.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 28-4: POSxCNT: POSITION COUNTER REGISTER ('X' = 1-3)

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| POSCNT<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| POSCNT<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| POSCNT<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| POSCNT<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **POSCNT<31:0>**: 32-bit Position Counter Register bits

The Operating mode of the position counter is controlled by the CCM bit in the QE1xCON register.

**Quadrature Count mode:** The QEA and QEB inputs are decoded to generate count pulses and direction information for controlling the position counter operation.

**External Count with External Up/Down mode:** The QEA/EXTCNT input is treated as an external count signal, and the QEB/DIR/GATE input provides the count direction information.

**External Count with External Gate mode:** The QEA/EXTCNT input is treated as an external count signal. If the GATEN bit in the QE1xCON register is equal to '1', the QEB/DIR/GATE input will gate the counter signal.

**Internal Timer mode:** The position counter uses PBCLK2 divided by the clock divider INTDIV as the count source.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 28-5: VELxCNT: VELOCITY COUNTER REGISTER ('X' = 1-3)

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELCNT<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELCNT<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELCNT<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELCNT<7:0>   |                |                |                |                |                |                |               |               |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

### bit 31-0 VELCNT<31:0>: 32-bit Velocity Counter bits

The velocity counter is automatically cleared after every processor read of the velocity counter. It is not reset by the index input or otherwise affected by any of the PIMOD<2:0> specified modes. The contents of the counter represents the distance traveled during the time between samples. Velocity equals the distance traveled per unit of time. The velocity counter can save the application software the trouble of performing 32-bit math operations between current and previous position counter values to calculate velocity. If the velocity counter rolls over from 0x7FFFFFFF to 0x80000000, or from 0x80000000 to 0x7FFFFFFF, an overflow/underflow condition is detected. If the VELOVIEN bit is set in the QEISTAT register, an interrupt will be generated.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 28-6: VELxHLD: VELOCITY HOLD REGISTER ('X' = 1-3)**

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELHLD<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELHLD<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELHLD<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| VELHLD<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **VELHLD<31:0>**: 32-bit Velocity Hold bits

When VELXCNT is read, the contents are captured at the same time into the VELxHLD register.

**REGISTER 28-7: INTxHLD: INTERVAL TIMER HOLD REGISTER ('X' = 1-3)**

| Bit Range     | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|---------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| INTHLD<31:24> |                |                |                |                |                |                |               |               |
| 23:16         | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| INTHLD<23:16> |                |                |                |                |                |                |               |               |
| 15:8          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| INTHLD<15:8>  |                |                |                |                |                |                |               |               |
| 7:0           | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
| INTHLD<7:0>   |                |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **INTHLD<31:0>**: 32-bit Index Counter Hold bits

When the next count pulse is detected, the current contents of the interval timer (INTxTMR) are transferred to the Interval Hold register (INTxHLD) and the interval timer is cleared and the process repeats.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 28-8: INDxCNT: INDEX COUNTER REGISTER ('X' = 1-3)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INDxCNT<31:24> |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INDxCNT<23:16> |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INDxCNT<15:8>  |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INDxCNT<7:0>   |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **IDXCNT<31:0>**: 32-bit Position Counter bits

**REGISTER 28-9: INTxTMR: INTERVAL TIMER REGISTER ('X' = 1-3)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INTTMR<31:24>  |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INTTMR<23:16>  |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INTTMR<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | INTTMR<7:0>    |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **INTTMR<31:0>**: 32-bit Interval Timer Counter bits

The INTxTMR register provides a means to measure the time between each decoded quadrature count pulse to yield improved velocity information. The interval timer should be set to run at a frequency chosen such that the counter does not overflow at the expected minimum operating speed of the motor. The interval timer is automatically cleared when a count pulse is detected. The timer then counts at the specified rate based on the setting of the INTDIV bit in the QE1xCON register.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 28-10: QEIXICC: QEIX INITIALIZE/CAPTURE/COMPARE REGISTER ('X' = 1-3)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | ICCH<31:24>    |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | ICCH<23:16>    |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | ICCH<15:8>     |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | ICCH<7:0>      |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **ICCH<31:0>**: 32-bit Initialize/Capture/Compare High bits

**REGISTER 28-11: QEIXCMPL: CAPTURE LOW REGISTER ('X' = 1-3)**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPL<31:24>    |                |                |                |                |                |               |               |
| 23:16     | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPL<23:16>    |                |                |                |                |                |               |               |
| 15:8      | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPL<15:8>     |                |                |                |                |                |               |               |
| 7:0       | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | CMPL<7:0>      |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
-n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **CMPL<31:0>**: 32-bit Compare Low Value bits



# PIC32MK GPG/MCJ with CAN FD Family

## 29.0 MOTOR CONTROL PWM MODULE

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 44. “Motor Control PWM (MCPWM)”** (DS60001393), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

The PIC32MK GPG/MCJ with CAN FD Family of devices support a dedicated Motor Control Pulse-Width Modulation (PWM) module with up to 24 outputs.

The Motor Control PWM module consists of the following major features:

- Two master time base modules with special event triggers
- PWM module input clock prescaler
- Two synchronization inputs
- Two synchronization outputs
- 9 PWM generators with complimentary output pairs
- Period, duty cycle, phase shift and dead time minimum resolution of 1/FSYSCLK in Edge-Aligned mode and 2/FSYSCLK minimum resolution in Center-Aligned mode
- Cycle by cycle fault recovery and latched fault modes
- PWM time-base capture upon current limit
- 10 fault input pins are available for faults and current limits
- Programmable analog-to-digital trigger with interrupt for each PWM pair
- Complementary PWM outputs
- Push-Pull PWM outputs
- Edge-Aligned PWM mode
- Center-Aligned PWM mode
- Variable Phase PWM mode
- Multi-Phase PWM mode

- Fixed-Off Time PWM mode
- Current Limit PWM mode
- Current Reset PWM mode
- PWMxH and PWMxL output override control
- PWMxH and PWMxL output pin swapping
- Chopping mode (also known as Gated mode)
- Dead time insertion
- Dead time compensation
- Enhanced Leading-Edge Blanking (LEB)
- 15 mA PWM pin output drive

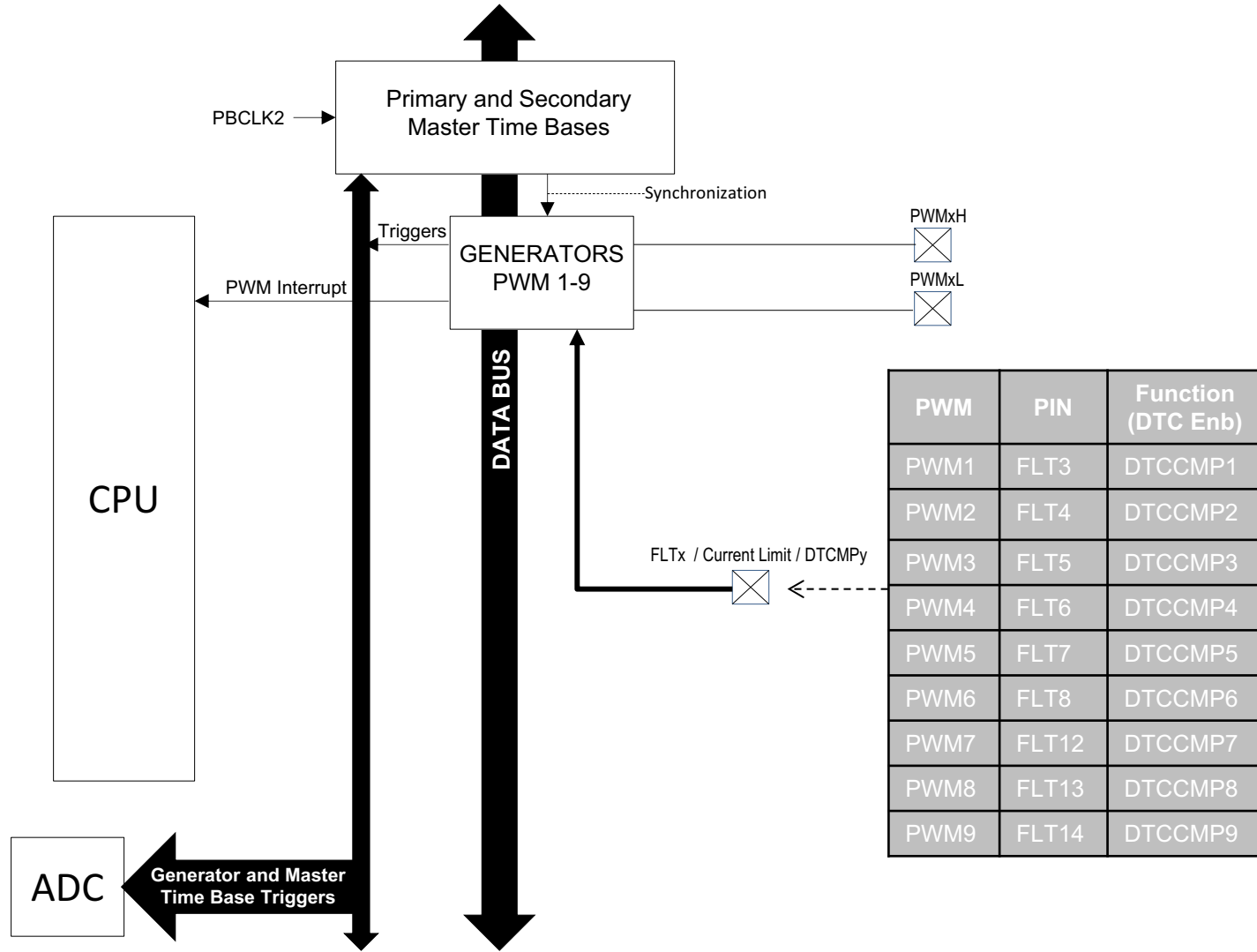
The Motor Control PWM module contains up to twelve PWM generators. Two master time base generators provide a synchronous signal as a common time base to synchronize the various PWM outputs. Each generator can operate independently or in synchronization with either of the two master time bases. The individual PWM outputs are available on the output pins of the device. The input Fault signals and current-limit signals, when enabled, can monitor and protect the system by placing the PWM outputs into a known “safe” state.

Each PWM can generate a trigger to the ADC module to sample the analog signal at a specific instance during the PWM period. In addition, the Motor Control PWM module also generates two Special Event Triggers to the ADC module based on the two master time bases.

PWM generators 1 through 9 have two outputs, PWMxH and PWMxL, brought out to the dedicated pins.

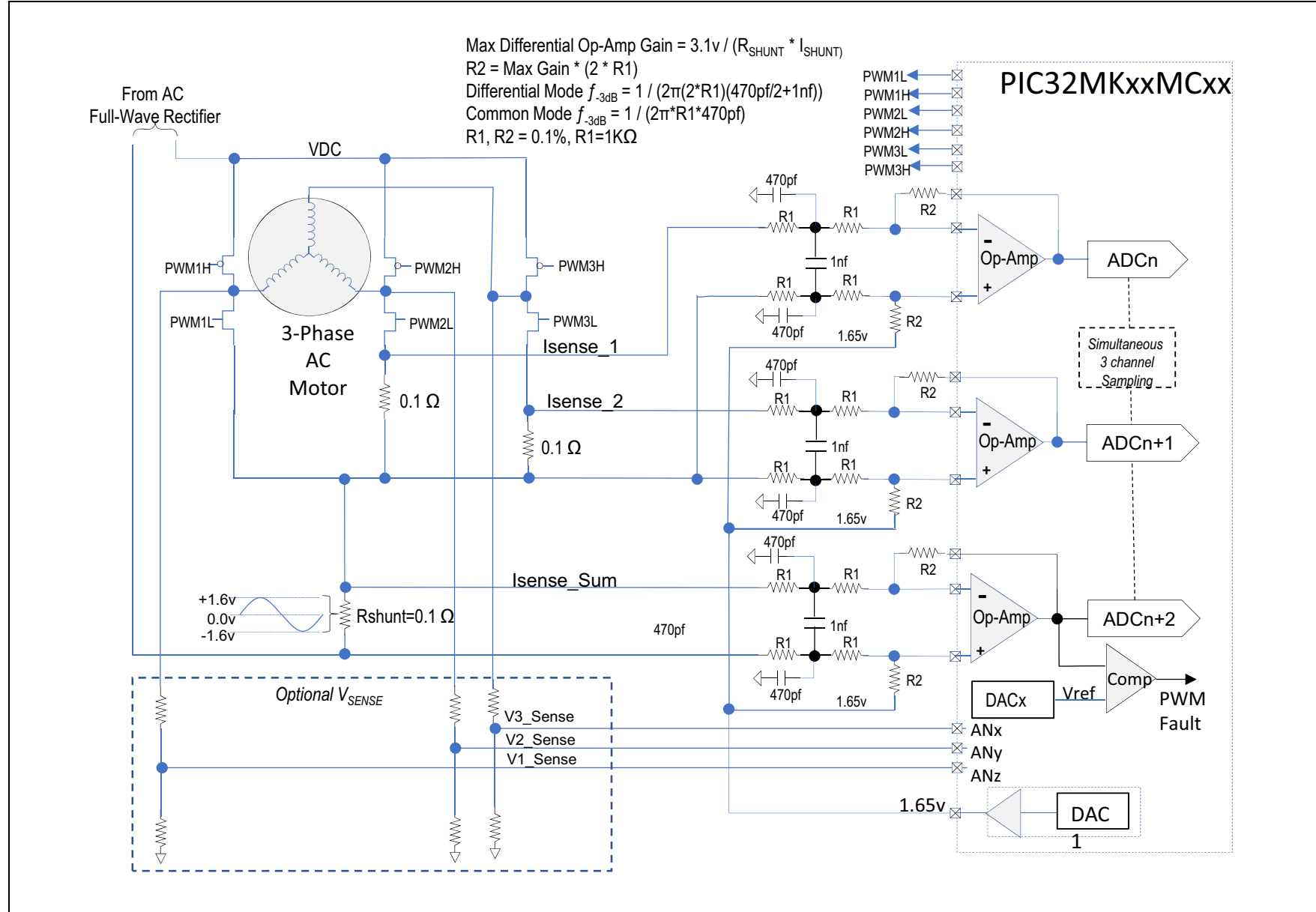
[Figure 29-1](#) illustrates an architectural overview of the Motor Control PWM module and its interconnection with the CPU and other peripherals.

**FIGURE 29-1: MOTOR CONTROL PWM MODULE ARCHITECTURAL OVERVIEW**



NOTE: Since DTCCMPx, (i.e. dead time compensation), FLTx and Current Limit share the same digital input pins, their availability is mutually exclusive with fault function as the highest priority.

**FIGURE 29-2: 3 PHASE AC MOTOR CONTROL EXAMPLE**



# PIC32MK GPG/MCJ with CAN FD Family

## 29.1 PWM Faults

The PWM module incorporates multiple external Fault inputs to include FLT1 and FLT2, which are remappable using the PPS feature, and FLT15, which has been implemented with Class B safety features, and is available on a fixed pin at reset for Fault detection.

Fault pins are selectable for active level (active high or low). FLT pins provide a safe and reliable way to shut down the PWM outputs, tri-state, when the Fault input is asserted. Therefore, the user should provide the necessary external pull-up or pull-down to disable the high or low side FETs in motor control applications.

### 29.1.1 PWM FAULTS AT RESET

During any reset event, the PWM module maintains ownership of the Class B fault FLT15. At reset, this fault is enabled in latched mode to guarantee the fail-safe power-up of the application. The application software must clear the PWM fault before enabling the High-Speed Motor Control PWM module. To clear the fault condition, the FLT15 pin must first be pulled low externally or the internal pull down resistor in the CNPDX register can be enabled.

**Note:** The Fault mode may be changed using the FLTMOD<1:0> bits (IOCONx<17:16>) regardless of the state of FLT15.

### EXAMPLE 29-1: PWM WRITE-PROTECTED REGISTER UNLOCK SEQUENCE

#### Untested Code - For Information Purposes Only

```
; In the default Reset state, the FLT15 pin must be pulled low externally to clear and disable
; the fault.
; Writing to IOCONx register requires unlock sequence
di    v1
ehb
mov   #0xFFFF, r3      ;Move desired IOCON4 register data to r3 register
mov   #0xabcd, r1       ;Load first unlock key to r1 register
mov   #0x4321, r2       ;Load second unlock key to r2 register
mov   r1, PWMKEY        ;Write first unlock key to PWMKEY register
mov   r2, PWMKEY        ;Write second unlock key to PWMKEY register
mov   r3, IOCON4        ;Write desired value to IOCON SFR for channel 4
mfc0  v0, c0_status
ori   v0, v0, 0x1
mtc0  v0, c0_status
ehb                          ;Re-enable Interrupts
```

### 29.1.2 WRITE-PROTECTED REGISTERS

Write protection is implemented for the IOCONx register. The write protection feature prevents any inadvertent writes. This protection feature can be controlled by the PWMLOCK Configuration bit (DEVCFG3<20>). The default state of the write protection feature is disabled (PWMLOCK = 1). The write protection feature can be enabled by configuring the PWMLOCK = 0.

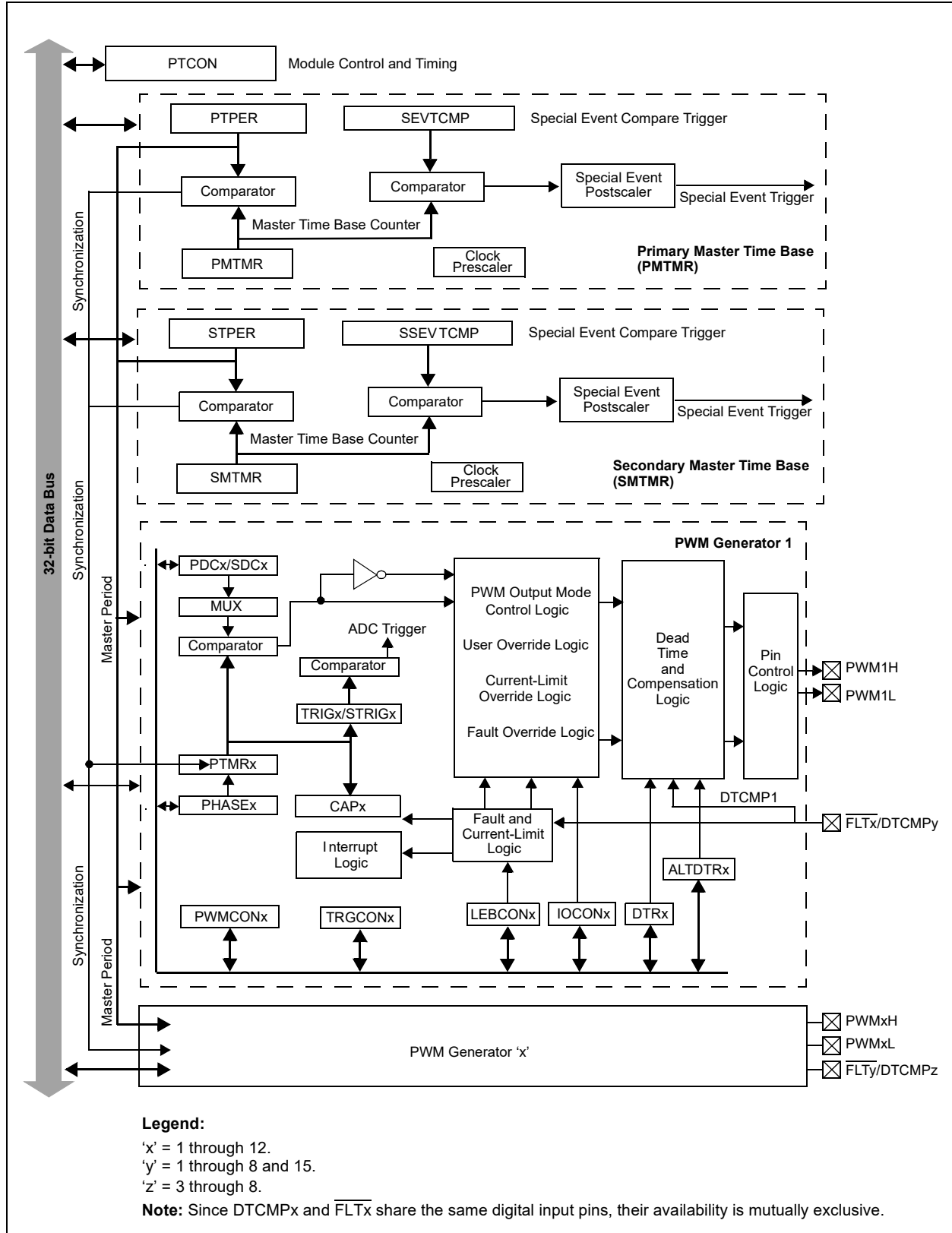
To gain write access, the application software must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation. The write access to the IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. Every write to the IOCONx register requires a prior unlock operation.

The unlocking sequence is described in [Example 29-1](#).

[Figure 29-3](#) shows the register interconnection diagram for the Motor Control PWM module.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 29-3: MOTOR CONTROL PWM MODULE REGISTER INTERCONNECTION DIAGRAM**



## 29.2 Motor Control PWM Control Registers

**TABLE 29-1: MCPWM REGISTER MAP**

| Virtual Address<br>(BF82_#) | Register Name | Bit Range | Bits           |         |            |         |           |        |        |              |              |              |        |             |             |      | All Resets |      |      |
|-----------------------------|---------------|-----------|----------------|---------|------------|---------|-----------|--------|--------|--------------|--------------|--------------|--------|-------------|-------------|------|------------|------|------|
|                             |               |           | 31/15          | 30/14   | 29/13      | 28/12   | 27/11     | 26/10  | 25/9   | 24/8         | 23/7         | 22/6         | 21/5   | 20/4        | 19/3        | 18/2 |            | 17/1 | 16/0 |
| A000                        | PTCON         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PTEN           | —       | PTSIDL     | SESTAT  | SEIEN     | PWMRDY | —      | —            | —            | PCLKDIV<2:0> |        |             | SEVTPS<3:0> |      |            | 0000 |      |
| A010                        | PTPER         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PTPER<15:0>    |         |            |         |           |        |        |              |              |              |        |             |             |      | 0020       |      |      |
| A020                        | SEVTCMP       | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | SEVTCMP<15:0>  |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A030                        | PMTMR         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PMTMR<15:0>    |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A040                        | STCON         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | —              | —       | —          | SSESTAT | SSEIEN    | —      | —      | —            | SCLKDIV<2:0> |              |        | SEVTPS<3:0> |             |      | 0000       |      |      |
| A050                        | STPER         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | STPER<15:0>    |         |            |         |           |        |        |              |              |              |        |             |             |      | 0020       |      |      |
| A060                        | SSEVTCMP      | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | SSEVTCMP<15:0> |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A070                        | SMTMR         | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | SMTMR<15:0>    |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A080                        | CHOP          | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | CHPCLKEN       | —       | —          | —       | —         | —      | —      | CHOPCLK<9:0> |              |              |        |             |             |      |            |      | 0000 |
| A090                        | PWMKEY        | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PWMKEY<15:0>   |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A0C0                        | PWMCN1        | 31:16     | FLTIF          | CLIF    | TRGIF      | PWMLIF  | PWMHIF    | —      | —      | —            | FLTIEN       | CLIEN        | TRGIEN | PWMLIEN     | PWMHIEN     | —    | —          | —    | 0000 |
|                             |               | 15:0      | FLTSTAT        | CLTSTAT | —          | —       | ECAM<1:0> | ITB    | —      | DTC<1:0>     | DTCP         | PTDIR        | MTBS   | —           | XPRES       | —    | —          | —    | 0000 |
| A0D0                        | IOCON1        | 31:16     | —              | —       | CLSRC<3:0> |         |           | CLPOL  | CLMOD  | —            | FLTSRC<3:0>  |              |        | FLTPOL      | FLTMOD<1:0> |      | 0078       |      |      |
|                             |               | 15:0      | PENH           | PENL    | POLH       | POLL    | PMOD<1:0> | OVRENH | OVRENL | OVRDAT<1:0>  | FLTDAT<1:0>  | CLDAT<1:0>   | SWAP   | OSYNC       | —           | —    | —          | 0000 |      |
| A0E0                        | PDC1          | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PDC<15:0>      |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A0F0                        | SDC1          | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | SDC<15:0>      |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A100                        | PHASE1        | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | PHASE<15:0>    |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |
| A110                        | DTR1          | 31:16     | —              | —       | —          | —       | —         | —      | —      | —            | —            | —            | —      | —           | —           | —    | —          | —    | 0000 |
|                             |               | 15:0      | DTR<15:0>      |         |            |         |           |        |        |              |              |              |        |             |             |      | 0000       |      |      |

Legend: '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register Name | Bit Range | Bits          |         |            |             |           |              |        |        |             |       |             |         |             |      |       | All Resets |      |
|-----------------------------|---------------|-----------|---------------|---------|------------|-------------|-----------|--------------|--------|--------|-------------|-------|-------------|---------|-------------|------|-------|------------|------|
|                             |               |           | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10        | 25/9   | 24/8   | 23/7        | 22/6  | 21/5        | 20/4    | 19/3        | 18/2 | 17/1  |            | 16/0 |
| A120                        | ALTDTR1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | ALTDTR<15:0>  |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A130                        | DTCOMP1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | COMP<13:0>    |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A140                        | TRIG1         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | TRGCMP<15:0>  |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A150                        | TRGCON1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           | STRGSEL<1:0> |        | DTM    | STRGIS      |       | —           | —       | —           | —    | —     | —          | 0000 |
| A160                        | STRIG1        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | STRGCMP<15:0> |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A170                        | CAP1          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | CAP<15:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A180                        | LEBCON1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN      | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
| A190                        | LEBDLY1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | LEB<11:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A1A0                        | AUXCON1       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | CHOPSEL<3:0>  |         |            |             |           |              |        |        |             |       |             | CHOPHEN | CHOPLEN     | 0000 |       |            |      |
| A1B0                        | PTMR1         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | TMR<15:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A1C0                        | PWMCN2        | 31:16     | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —            | —      | —      | FLTIEN      | CLIEN | TRGIEN      | PWMLIEN | PWMHIEN     | —    | —     | —          | 0000 |
|                             |               | 15:0      | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |              | ITB    | —      | DTC<1:0>    |       | DTCP        | PTDIR   | MTBS        | —    | XPRES | —          | 0000 |
| A1D0                        | IOCON2        | 31:16     | —             | —       | CLSRC<3:0> |             |           | CLPOL        | CLMOD  | —      | FLTSRC<3:0> |       |             | FLTPOL  | FLTMOD<1:0> |      | —     | —          | 0078 |
|                             |               | 15:0      | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |              | OVRENH | OVRENL | OVRDAT<1:0> |       | FLTDAT<1:0> |         | CLDAT<1:0>  |      | SWAP  | OSYNC      | 0000 |
| A1E0                        | PDC2          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | PDC<15:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A1F0                        | SDC2          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | SDC<15:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A200                        | PHASE2        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | PHASE<15:0>   |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A210                        | DTR2          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | DTR<15:0>     |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |
| A220                        | ALTDTR2       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —     | —           | —       | —           | —    | —     | —          | 0000 |
|                             |               | 15:0      | ALTDTR<15:0>  |         |            |             |           |              |        |        |             |       |             |         |             |      |       | 0000       |      |

Legend: '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register<br>Name | Bit<br>Range | Bits          |         |            |             |           |         |              |        |             |        |              |         |             |         |         | All<br>Resets |      |
|-----------------------------|------------------|--------------|---------------|---------|------------|-------------|-----------|---------|--------------|--------|-------------|--------|--------------|---------|-------------|---------|---------|---------------|------|
|                             |                  |              | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10   | 25/9         | 24/8   | 23/7        | 22/6   | 21/5         | 20/4    | 19/3        | 18/2    | 17/1    |               | 16/0 |
| A230                        | DTCOMP2          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | COMP<13:0>    |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A240                        | TRIG2            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TRGCMP<15:0>  |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A250                        | TRGCON2          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           |         | STRGSEL<1:0> |        | DTM         | STRGIS | —            | —       | —           | —       | —       | —             | 0000 |
| A260                        | STRIG2           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | STRGCMP<15:0> |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A270                        | CAP2             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | CAP<15:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A280                        | LEBCON2          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
| A290                        | LEBDLY2          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | LEB<11:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A2A0                        | AUXCON2          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | CHOPSEL<3:0> |         |             | CHOPHEN | CHOPLEN | 0000          |      |
| A2B0                        | PTMR2            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TMR<15:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A2C0                        | PWMCON3          | 31:16        | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —       | —            | —      | FLTIEN      | CLIEEN | TRGIEN       | PWMLIEN | PWMHIEN     | —       | —       | —             | 0000 |
|                             |                  | 15:0         | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |         | ITB          | —      | DTC<1:0>    |        | DTCP         | PTDIR   | MTBS        | —       | XPRES   | —             | 0000 |
| A2D0                        | IOCON3           | 31:16        | —             | —       | CLSRC<3:0> |             |           | CLPOL   | CLMOD        | —      | FLTSRC<3:0> |        |              | FLTPOL  | FLTMOD<1:0> |         | 0078    |               |      |
|                             |                  | 15:0         | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |         | OVRENH       | OVRENL | OVRDAT<1:0> |        | FLTDAT<1:0>  |         | CLDAT<1:0>  | SWAP    | OSYNC   | 0000          |      |
| A2E0                        | PDC3             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | 0000          |      |
|                             |                  | 15:0         | PDC<15:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A2F0                        | SDC3             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | SDC<15:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A300                        | PHASE3           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | PHASE<15:0>   |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A310                        | DTR3             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | DTR<13:0>     |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A320                        | ALTDTR3          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | ALTDTR<15:0>  |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |
| A330                        | DTCOMP3          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —       | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | COMP<13:0>    |         |            |             |           |         |              |        |             |        |              |         |             |         |         | 0000          |      |

Legend: '—' = unimplemented; read as '0'.



TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register Name | Bit Range | Bits          |         |            |             |           |              |        |             |             |              |        |            |             |         |       | All Resets |
|-----------------------------|---------------|-----------|---------------|---------|------------|-------------|-----------|--------------|--------|-------------|-------------|--------------|--------|------------|-------------|---------|-------|------------|
|                             |               |           | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10        | 25/9   | 24/8        | 23/7        | 22/6         | 21/5   | 20/4       | 19/3        | 18/2    | 17/1  |            |
| A340                        | TRIG3         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | TRGCMP<15:0>  |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A350                        | TRGCON3       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           | STRGSEL<1:0> |        | DTM         | STRGIS      | —            | —      | —          | —           | —       | —     | 0000       |
| A360                        | STRIG3        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | STRGCMP<15:0> |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A370                        | CAP3          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | CAP<15:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A380                        | LEBCON3       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN      | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
| A390                        | LEBDLY3       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | LEB<11:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A3A0                        | AUXCON3       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | —             | —       | —          | —           | —         | —            | —      | —           | —           | CHOPSEL<3:0> |        |            | CHOPHEN     | CHOPLEN | 0000  |            |
| A3B0                        | PTMR3         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | TMR<15:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A3C0                        | PWMCON4       | 31:16     | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —            | —      | —           | FLTIEN      | CLIEEN       | TRGIEN | PWMLIEN    | PWMHIEN     | —       | —     | 0000       |
|                             |               | 15:0      | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |              | ITB    | —           | DTC<1:0>    |              | DTCP   | PTDIR      | MTBS        | —       | XPRES | —          |
| A3D0                        | IOCON4        | 31:16     | —             | —       | CLSRC<3:0> |             |           | CLPOL        | CLMOD  | —           | FLTSRC<3:0> |              |        | FLTPOL     | FLTMOD<1:0> |         | 0078  |            |
|                             |               | 15:0      | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> | OVRENH       | OVRENL | OVRDAT<1:0> |             | FLTDAT<1:0>  |        | CLDAT<1:0> |             | SWAP    | OSYNC |            |
| A3E0                        | PDC4          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | PDC<15:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A3F0                        | SDC4          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | SDC<15:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A400                        | PHASE4        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | PHASE<15:0>   |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A410                        | DTR4          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | DTR<15:0>     |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A420                        | ALTDTR4       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | ALTDTR<15:0>  |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A430                        | DTCOMP4       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | COMP<13:0>    |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |
| A440                        | TRIG4         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —           | —           | —            | —      | —          | —           | —       | —     | 0000       |
|                             |               | 15:0      | TRGCMP<15:0>  |         |            |             |           |              |        |             |             |              |        |            |             |         |       | 0000       |

Legend: '—' = unimplemented; read as '0'.

**TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF82_#) | Register<br>Name | Bit<br>Range | Bits          |         |            |             |           |         |              |        |             |        |              |        |             |         |        | All<br>Resets |      |      |
|-----------------------------|------------------|--------------|---------------|---------|------------|-------------|-----------|---------|--------------|--------|-------------|--------|--------------|--------|-------------|---------|--------|---------------|------|------|
|                             |                  |              | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10   | 25/9         | 24/8   | 23/7        | 22/6   | 21/5         | 20/4   | 19/3        | 18/2    | 17/1   |               | 16/0 |      |
| A450                        | TRGCON4          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           |         | STRGSEL<1:0> |        | DTM         | STRGIS | —            | —      | —           | —       | —      | —             | 0000 |      |
| A460                        | STRIG4           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | STRGCMP<15:0> |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A470                        | CAP4             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | CAP<15:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A480                        | LEBCON4          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
| A490                        | LEBDLY4          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | LEB<11:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A4A0                        | AUXCON4          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | CHOPSEL<3:0> |        |             | CHOPHEN | CHOPLN | 0000          |      |      |
| A4B0                        | PTMR4            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | TMR<15:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A4C0                        | PWMCON5          | 31:16        | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —       | —            | —      | —           | FLTIE  | CLIE         | TRGIE  | PWMLIE      | PWMHIE  | —      | —             | —    | 0000 |
|                             |                  | 15:0         | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |         | ITB          | —      | DTC<1:0>    |        | DTCP         | PTDIR  | MTBS        | —       | XPRES  | —             | 0000 |      |
| A4D0                        | IOCON5           | 31:16        | —             | —       | CLSRC<3:0> |             |           | CLPOL   | CLMOD        | —      | FLTSRC<3:0> |        |              | FLTPOL | FLTMOD<1:0> |         | 0078   |               |      |      |
|                             |                  | 15:0         | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |         | OVRENH       | OVRENL | OVRDAT<1:0> |        | FLTDAT<1:0>  |        | CLDAT<1:0>  |         | SWAP   | OSYNC         | 0000 |      |
| A4E0                        | PDC5             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | PDC<15:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A4F0                        | SDC5             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | SDC<15:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A500                        | PHASE5           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | PHASE<15:0>   |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A510                        | DTR5             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | DTR<15:0>     |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A520                        | ALTDTR5          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | ALTDTR<15:0>  |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A530                        | DTCOMP5          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | COMP<13:0>    |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A540                        | TRIG5            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | TRGCMP<15:0>  |         |            |             |           |         |              |        |             |        |              |        |             |         |        | 0000          |      |      |
| A550                        | TRGCON5          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —      | —           | —       | —      | —             | 0000 |      |
|                             |                  | 15:0         | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           |         | STRGSEL<1:0> |        | DTM         | STRGIS | —            | —      | —           | —       | —      | —             | 0000 |      |

**Legend:** '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register Name | Bit Range | Bits          |         |            |             |           |              |        |        |             |      |              |        |             |         |         | All Resets |      |
|-----------------------------|---------------|-----------|---------------|---------|------------|-------------|-----------|--------------|--------|--------|-------------|------|--------------|--------|-------------|---------|---------|------------|------|
|                             |               |           | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10        | 25/9   | 24/8   | 23/7        | 22/6 | 21/5         | 20/4   | 19/3        | 18/2    | 17/1    |            | 16/0 |
| A560                        | STRIG5        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | STRGCMP<15:0> |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A570                        | CAP5          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | CAP<15:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A580                        | LEBCON5       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN      | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
| A590                        | LEBDLY5       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | LEB<11:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A5A0                        | AUXCON5       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | CHOPSEL<3:0> |        |             | CHOPHEN | CHOPLEN | 0000       |      |
| A5B0                        | PTMR5         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | TMR<15:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A5C0                        | PWMCON6       | 31:16     | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |              | ITB    | —      | DTC<1:0>    |      | DTCP         | PTDIR  | MTBS        | —       | XPRES   | —          | 0000 |
| A5D0                        | IOCON6        | 31:16     | —             | —       | CLSRC<3:0> |             |           | CLPOL        | CLMOD  | —      | FLTSRC<3:0> |      |              | FLTPOL | FLTMOD<1:0> |         | —       | —          | 0078 |
|                             |               | 15:0      | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |              | OVRENH | OVRENL | OVRDAT<1:0> |      | FLTDAT<1:0>  |        | CLDAT<1:0>  |         | SWAP    | OSYNC      | 0000 |
| A5E0                        | PDC6          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | PDC<15:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A5F0                        | SDC6          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | SDC<15:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A600                        | PHASE6        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | PHASE<15:0>   |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A610                        | DTR6          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | DTR<15:0>     |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A620                        | ALTDTR6       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | ALTDTR<15:0>  |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A630                        | DTCOMP6       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | COMP<13:0>    |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A640                        | TRIG6         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | TRGCMP<15:0>  |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |
| A650                        | TRGCON6       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           | STRGSEL<1:0> |        | DTM    | STRGIS      | —    | —            | —      | —           | —       | —       | —          | 0000 |
| A660                        | STRIG6        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —      | —           | —       | —       | —          | 0000 |
|                             |               | 15:0      | STRGCMP<15:0> |         |            |             |           |              |        |        |             |      |              |        |             |         |         | 0000       |      |

Legend: '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register<br>Name | Bit<br>Range | Bits          |         |            |             |           |         |              |        |             |        |              |          |             |         |         | All<br>Resets |      |
|-----------------------------|------------------|--------------|---------------|---------|------------|-------------|-----------|---------|--------------|--------|-------------|--------|--------------|----------|-------------|---------|---------|---------------|------|
|                             |                  |              | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10   | 25/9         | 24/8   | 23/7        | 22/6   | 21/5         | 20/4     | 19/3        | 18/2    | 17/1    |               | 16/0 |
| A670                        | CAP6             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | CAP<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A680                        | LEBCON6          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
| A690                        | LEBDLY6          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | LEB<11:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A6A0                        | AUXCON6          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | CHOPSEL<3:0> |          |             | CHOPHEN | CHOPLEN | 0000          |      |
| A6B0                        | PTMR6            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TMR<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A6C0                        | PWMCON7          | 31:16        | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —       | —            | —      | FLTIEN      | CLIEEN | TRGIEEN      | PWMLIEEN | PWMHIEEN    | —       | —       | —             | 0000 |
|                             |                  | 15:0         | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |         | ITB          | —      | DTC<1:0>    |        | DTCP         | PTDIR    | MTBS        | —       | XPRES   | —             | 0000 |
| A6D0                        | IOCON7           | 31:16        | —             | —       | CLSRC<3:0> |             |           | CLPOL   | CLMOD        | —      | FLTSRC<3:0> |        |              | FLTPOL   | FLTMOD<1:0> |         | 0078    |               |      |
|                             |                  | 15:0         | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |         | OVRENH       | OVRENL | OVRDAT<1:0> |        | FLTDAT<1:0>  |          | CLDAT<1:0>  |         | SWAP    | OSYNC         | 0000 |
| A6E0                        | PDC7             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | PDC<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A6F0                        | SDC7             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | SDC<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A700                        | PHASE7           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | PHASE<15:0>   |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A710                        | DTR7             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | DTR<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A720                        | ALTDTR7          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | ALTDTR<15:0>  |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A730                        | DTCOMP7          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | COMP<13:0>    |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A740                        | TRIG7            | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TRGCMP<15:0>  |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A750                        | TRGCON7          | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           |         | STRGSEL<1:0> |        | DTM         | STRGIS | —            | —        | —           | —       | —       | —             | 0000 |
| A760                        | STRIG7           | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | STRGCMP<15:0> |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |
| A770                        | CAP7             | 31:16        | —             | —       | —          | —           | —         | —       | —            | —      | —           | —      | —            | —        | —           | —       | —       | —             | 0000 |
|                             |                  | 15:0         | CAP<15:0>     |         |            |             |           |         |              |        |             |        |              |          |             |         |         | 0000          |      |

Legend: '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address (BF82_#) | Register Name | Bit Range | Bits          |         |            |             |           |              |        |        |             |      |              |            |             |         |        | All Resets |      |
|--------------------------|---------------|-----------|---------------|---------|------------|-------------|-----------|--------------|--------|--------|-------------|------|--------------|------------|-------------|---------|--------|------------|------|
|                          |               |           | 31/15         | 30/14   | 29/13      | 28/12       | 27/11     | 26/10        | 25/9   | 24/8   | 23/7        | 22/6 | 21/5         | 20/4       | 19/3        | 18/2    | 17/1   |            | 16/0 |
| A780                     | LEBCON7       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN      | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
| A790                     | LEBDLY7       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | —             | —       | —          | —           | LEB<11:0> |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A7A0                     | AUXCON7       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | CHOPSEL<3:0> |            |             | CHOPHEN | CHOPLN | 0000       |      |
| A7B0                     | PTMR7         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | TMR<15:0>     |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A7C0                     | PWMCON8       | 31:16     | FLTIF         | CLIF    | TRGIF      | PWMLIF      | PWMHIF    | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | FLTSTAT       | CLTSTAT | —          | —           | ECAM<1:0> |              | ITB    | —      | DTC<1:0>    |      | DTCP         | PTDIR      | MTBS        | —       | XPRES  | —          | 0000 |
| A7D0                     | IOCON8        | 31:16     | —             | —       | CLSRC<3:0> |             |           | CLPOL        | CLMOD  | —      | FLTSRC<3:0> |      |              | FLTPOL     | FLTMOD<1:0> |         | 0078   |            |      |
|                          |               | 15:0      | PENH          | PENL    | POLH       | POLL        | PMOD<1:0> |              | OVRENH | OVRENL | OVRDAT<1:0> |      | FLTDAT<1:0>  | CLDAT<1:0> |             | SWAP    | OSYNC  | 0000       |      |
| A7E0                     | PDC8          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | PDC<15:0>     |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A7F0                     | SDC8          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | SDC<15:0>     |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A800                     | PHASE8        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | PHASE<15:0>   |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A810                     | DTR8          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | DTR<15:0>     |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A820                     | ALTDTR8       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | ALTDTR<15:0>  |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A830                     | DTCOMP8       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | COMP<13:0>    |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A840                     | TRIG8         | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | TRGCMP<15:0>  |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A850                     | TRGCON8       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | TRGDIV<3:0>   |         |            | TRGSEL<1:0> |           | STRGSEL<1:0> |        | DTM    | STRGIS      | —    | —            | —          | —           | —       | —      | —          | 0000 |
| A860                     | STRIG8        | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | STRGCMP<15:0> |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A870                     | CAP8          | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | CAP<15:0>     |         |            |             |           |              |        |        |             |      |              |            |             |         |        | 0000       |      |
| A880                     | LEBCON8       | 31:16     | —             | —       | —          | —           | —         | —            | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |
|                          |               | 15:0      | PHR           | PHF     | PLR        | PLF         | FLTLEBEN  | CLLEBEN      | —      | —      | —           | —    | —            | —          | —           | —       | —      | —          | 0000 |

Legend: '—' = unimplemented; read as '0'.

**TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)**

| Virtual Address<br>(BF82_#) | Register<br>Name | Bit<br>Range | Bits          |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | All<br>Resets |
|-----------------------------|------------------|--------------|---------------|---------|------------|--------|-------------|--------------|--------|-------------|-------------|-------|------------|--------|-------------|---------|---------|------|---------------|
|                             |                  |              | 31/15         | 30/14   | 29/13      | 28/12  | 27/11       | 26/10        | 25/9   | 24/8        | 23/7        | 22/6  | 21/5       | 20/4   | 19/3        | 18/2    | 17/1    | 16/0 |               |
| A890                        | LEBDLY8          | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | —    | 0000          |
|                             |                  | 15:0         | LEB<11:0>     |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A8A0                        | AUXCON8          | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | —    | 0000          |
|                             |                  | 15:0         | CHOPSEL<3:0>  |         |            |        |             |              |        |             |             |       |            |        |             | CHOPHEN | CHOPLEN | 0000 |               |
| A8B0                        | PTMR8            | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | —    | 0000          |
|                             |                  | 15:0         | TMR<15:0>     |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A8C0                        | PWMCON9          | 31:16        | FLTIF         | CLIF    | TRGIF      | PWMLIF | PWMHIF      | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | —    | 0000          |
|                             |                  | 15:0         | FLTSTAT       | CLTSTAT | —          | —      | ECAM<1:0>   | ITB          | —      | DTC<1:0>    | DTCP        | PTDIR | MTBS       | —      | XPRES       | —       | —       | —    | 0000          |
| A8D0                        | IOCON9           | 31:16        | —             | —       | CLSRC<3:0> |        |             | CLPOL        | CLMOD  | —           | FLTSRC<3:0> |       |            | FLTPOL | FLTMOD<1:0> |         | 0078    |      |               |
|                             |                  | 15:0         | PENH          | PENL    | POLH       | POLL   | PMOD<1:0>   | OVRENH       | OVRENL | OVRDAT<1:0> | FLTDAT<1:0> |       | CLDAT<1:0> |        | SWAP        | OSYNC   | 0000    |      |               |
| A8E0                        | PDC9             | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | PDC<15:0>     |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A8F0                        | SDC9             | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | SDC<15:0>     |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A900                        | PHASE9           | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | PHASE<15:0>   |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A910                        | DTR9             | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | DTR<15:0>     |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A920                        | ALTDTR9          | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | ALTDTR<15:0>  |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A930                        | DTCOMP9          | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | COMP<13:0>    |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A940                        | TRIG9            | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | TRGCMP<15:0>  |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |
| A950                        | TRGCON9          | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | TRGDIV<3:0>   |         |            |        | TRGSEL<1:0> | STRGSEL<1:0> | DTM    | STRGIS      | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
| A960                        | STRIG9           | 31:16        | —             | —       | —          | —      | —           | —            | —      | —           | —           | —     | —          | —      | —           | —       | —       | 0000 |               |
|                             |                  | 15:0         | STRGCMP<15:0> |         |            |        |             |              |        |             |             |       |            |        |             |         |         |      | 0000          |

**Legend:** '—' = unimplemented; read as '0'.

TABLE 29-1: MCPWM REGISTER MAP (CONTINUED)

| Virtual Address<br>(BF82_#) | Register Name | Bit Range | Bits      |       |       |       |          |         |      |      |      |      |              |      |      |         |         | All Resets |      |
|-----------------------------|---------------|-----------|-----------|-------|-------|-------|----------|---------|------|------|------|------|--------------|------|------|---------|---------|------------|------|
|                             |               |           | 31/15     | 30/14 | 29/13 | 28/12 | 27/11    | 26/10   | 25/9 | 24/8 | 23/7 | 22/6 | 21/5         | 20/4 | 19/3 | 18/2    | 17/1    |            | 16/0 |
| A970                        | CAP9          | 31:16     | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
|                             |               | 15:0      | CAP<15:0> |       |       |       |          |         |      |      |      |      |              |      |      |         |         | 0000       |      |
| A980                        | LEBCON9       | 31:16     | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
|                             |               | 15:0      | PHR       | PHF   | PLR   | PLF   | FLTLEBEN | CLLEBEN | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
| A990                        | LEBDLY9       | 31:16     | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
|                             |               | 15:0      | LEB<11:0> |       |       |       |          |         |      |      |      |      |              |      |      |         |         | 0000       |      |
| A9A0                        | AUXCON9       | 31:16     | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
|                             |               | 15:0      | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | CHOPSEL<3:0> |      |      | CHOPHEN | CHOPLEN | 0000       |      |
| A9B0                        | PTMR9         | 31:16     | —         | —     | —     | —     | —        | —       | —    | —    | —    | —    | —            | —    | —    | —       | —       | —          | 0000 |
|                             |               | 15:0      | TMR<15:0> |       |       |       |          |         |      |      |      |      |              |      |      |         |         | 0000       |      |

Legend: '—' = unimplemented; read as '0'.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-1: PTCN: PWM PRIMARY TIME BASE CONTROL REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6           | Bit<br>29/21/13/5 | Bit<br>28/20/12/4     | Bit<br>27/19/11/3          | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-----------------------------|-------------------|-----------------------|----------------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0                         | U-0               | U-0                   | U-0                        | U-0               | U-0              | U-0              |
|           | —                 | —                           | —                 | —                     | —                          | —                 | —                | —                |
| 23:16     | U-0               | U-0                         | U-0               | U-0                   | U-0                        | U-0               | U-0              | U-0              |
|           | —                 | —                           | —                 | —                     | —                          | —                 | —                | —                |
| 15:8      | R/W-0             | U-0                         | R/W-0             | HS/HC-0               | R/W-0                      | HS/HC-0           | U-0              | U-0              |
|           | PTEN              | —                           | PTSIDL            | SESTAT <sup>(1)</sup> | SEIEN <sup>(3)</sup>       | PWMRDY            | —                | —                |
| 7:0       | U-0               | R/W-0                       | R/W-0             | R/W-0                 | R/W-0                      | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | PCLKDIV<2:0> <sup>(2)</sup> |                   |                       | SEVTPS<3:0> <sup>(2)</sup> |                   |                  |                  |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **PTEN:** PWM Module Enable bit

1 = PWM module is enabled  
0 = PWM module is disabled

**Note:** Many of the PWM registers and/or bits as designated, do not allow updates once a PWM module is enabled. Therefore, it is recommended that the user application initialize all required PWM registers before setting the PTEN bit equal to '1'.

bit 14 **Unimplemented:** Read as '0'

bit 13 **PTSIDL:** PWM Time Base Stop in Idle Mode bit

1 = PWM time base halts in CPU Idle mode  
0 = PWM time base runs in CPU Idle mode

bit 12 **SESTAT:** Special Event Interrupt Status bit<sup>(1)</sup>

1 = Special Event Interrupt is pending  
0 = Special Event Interrupt is not pending

bit 11 **SEIEN:** Special Event Interrupt Enable bit

1 = Special Event Interrupt is enabled  
0 = Special Event Interrupt is disabled

bit 10 **PWMRDY:** PWM Module Status bit

1 = PWM module is ready and operation has begun  
0 = PWM module is not ready

bit 9-7 **Unimplemented:** Read as '0'

bit 6-4 **PCLKDIV<2:0>:** Primary PWM Input Clock Prescaler bits<sup>(2)</sup>

111 = Divide by 128, PWM resolution = 128/FSYSCLK  
110 = Divide by 64, PWM resolution = 64/FSYSCLK

•  
•  
•

000 = Divide by 1, PWM resolution = 1/FSYSCLK (power-on default)

**Note 1:** The SESTAT bit is cleared by clearing the SEIEN bit and the corresponding bit in the IFSx register.

**2:** The SEVTPS<3:0> bits should be changed only when the PTEN bit (PTCON<15>) = 0.

**3:** To clear the Primary Special Event Interrupt the user application must do the following:

- 1) Clear the SEIEN bit by setting it to '0'.
- 2) Clear the Primary Special Event Interrupt flag by setting IFS5<11> = 0.
- 3) Re-enabling the PTCN register by setting the SEIEN equal to '1' if desired.

The user application will not be able to clear the Primary Special Event Interrupt flag as long as the SEIEN bit is equal to '1'.



# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 29-1: PTCN: PWM PRIMARY TIME BASE CONTROL REGISTER (CONTINUED)

bit 3-0      **SEVTPS<3:0>**: PWM Special Event Trigger Output Postscaler Select bits<sup>(2)</sup>

- 1111 = 1:16 postscaler generates Special Event trigger at every 16th compare match event
- 
- 
- 
- 0001 = 1:2 postscaler generates Special Event trigger at every second compare match event
- 0000 = 1:1 postscaler generates Special Event trigger at every compare match event

- Note 1:** The SESTAT bit is cleared by clearing the SEIEN bit and the corresponding bit in the IFSx register.
- 2:** The SEVTPS<3:0> bits should be changed only when the PTEN bit (PTCON<15>) = 0.
- 3:** To clear the Primary Special Event Interrupt the user application must do the following:
- 1) Clear the SEIEN bit by setting it to '0'.
  - 2) Clear the Primary Special Event Interrupt flag by setting IFS5<11> = 0.
  - 3) Re-enabling the PTCN register by setting the SEIEN equal to '1' if desired.
- The user application will not be able to clear the Primary Special Event Interrupt flag as long as the SEIEN bit is equal to '1'.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-2: PTPER: PRIMARY MASTER TIME BASE PERIOD REGISTER

| Bit Range | Bit<br>31/23/15/7            | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2    | Bit<br>25/17/9/1     | Bit<br>24/16/8/0     |
|-----------|------------------------------|-------------------|-------------------|-------------------|-------------------|----------------------|----------------------|----------------------|
| 31:24     | U-0                          | U-0               | U-0               | U-0               | U-0               | U-0                  | U-0                  | U-0                  |
|           | —                            | —                 | —                 | —                 | —                 | —                    | —                    | —                    |
| 23:16     | U-0                          | U-0               | U-0               | U-0               | U-0               | U-0                  | U-0                  | U-0                  |
|           | —                            | —                 | —                 | —                 | —                 | —                    | —                    | —                    |
| 15:8      | R/W-0                        | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0                | R/W-0                | R/W-0                |
|           | PTPER<15:8> <sup>(1,2)</sup> |                   |                   |                   |                   |                      |                      |                      |
| 7:0       | R/W-0                        | R/W-0             | R/W-1             | R/W-0             | R/W-0             | R/W-0 <sup>(3)</sup> | R/W-0 <sup>(3)</sup> | R/W-0 <sup>(3)</sup> |
|           | PTPER<7:0> <sup>(1,2)</sup>  |                   |                   |                   |                   |                      |                      |                      |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **PTPER<15:0>:** Primary Master Time Base Period Value bits<sup>(1,2,4)</sup>

**Note 1:** Minimum LSb = 1/FSYSCLK.

**2:** Minimum value is 0x0008.

**3:** If a period value is lesser than 0x0008 is chosen, the internal hardware forcefully sets the period to a minimum value of 0x0008.

**4:**  $PTPER = (FSYSCLK / (F_{PWM} * PTCN<PCLKDIV>))$   
 $F_{PWM}$  = User Desired PWM Frequency

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-3: SEVTCMP: PWM PRIMARY SPECIAL EVENT COMPARE REGISTER**

| Bit Range | Bit 31/23/15/7                 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|--------------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0                            | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R/W-0                          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | SEVTCMP<15:8> <sup>(1,2)</sup> |                |                |                |                |                |               |               |
| 7:0       | R/W-0                          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0          | R/W-0         | R/W-0         |
|           | SEVTCMP<7:0> <sup>(1,2)</sup>  |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                                      W = Writable bit                                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                                      '1' = Bit is set                                      '0' = Bit is cleared                                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **SEVTCMP<15:0>:** Special Event Compare Count Value bits<sup>(1,2)</sup>

The special event trigger allows analog-to-digital conversions to be synchronized to the master PWM time base. The analog-to-digital sampling and conversion time may be programmed to occur at any point within the PWM period.

**Note 1:** Minimum LSb = 1/FSYSCLK.

**2:** To trigger at the period boundary, set the SEVTCMP bit to 0x0 and not the PTPER period value.

**REGISTER 29-4: PMTMR: PRIMARY MASTER TIME BASE TIMER REGISTER**

| Bit Range | Bit 31/23/15/7             | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | U-0                        | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                          | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | U-0                        | U-0            | U-0            | U-0            | U-0            | U-0            | U-0           | U-0           |
|           | —                          | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | PMTMR<15:8> <sup>(1)</sup> |                |                |                |                |                |               |               |
| 7:0       | R-0                        | R-0            | R-0            | R-0            | R-0            | R-0            | R-0           | R-0           |
|           | PMTMR<7:0> <sup>(1)</sup>  |                |                |                |                |                |               |               |

**Legend:**

R = Readable bit                                      W = Writable bit                                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                                      '1' = Bit is set                                      '0' = Bit is cleared                                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **PMTMR<15:0>:** Primary Master Time Base Timer Value bits<sup>(1)</sup>

This timer increments with each PWM clock until the PTPER value is reached.

**Note 1:** LSb = 1/FSYSCLK.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-5: STCON: SECONDARY MASTER TIME BASE CONTROL REGISTER**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6           | Bit<br>29/21/13/5 | Bit<br>28/20/12/4      | Bit<br>27/19/11/3          | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-----------------------------|-------------------|------------------------|----------------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0                         | U-0               | U-0                    | U-0                        | U-0               | U-0              | U-0              |
|           | —                 | —                           | —                 | —                      | —                          | —                 | —                | —                |
| 23:16     | U-0               | U-0                         | U-0               | U-0                    | U-0                        | U-0               | U-0              | U-0              |
|           | —                 | —                           | —                 | —                      | —                          | —                 | —                | —                |
| 15:8      | U-0               | U-0                         | U-0               | HS/HC-0                | R/W-0                      | U-0               | U-0              | U-0              |
|           | —                 | —                           | —                 | SSESTAT <sup>(1)</sup> | SSEIEN <sup>(3)</sup>      | —                 | —                | —                |
| 7:0       | U-0               | R/W-0                       | R/W-0             | R/W-0                  | R/W-0                      | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | SCLKDIV<2:0> <sup>(2)</sup> |                   |                        | SEVTPS<3:0> <sup>(2)</sup> |                   |                  |                  |

**Legend:**

R = Readable bit                                  W = Writable bit                                  U = Unimplemented bit, read as '0'  
 -n = Value at POR                                  '1' = Bit is set                                  '0' = Bit is cleared                                  x = Bit is unknown

- bit 31-13    **Unimplemented:** Read as '0'
- bit 12        **SSESTAT:** Secondary Special Event Interrupt Status bit<sup>(1)</sup>
  - 1 = Secondary Special Event Interrupt is pending
  - 0 = Secondary Special Event Interrupt is not pending
- bit 11        **SSEIEN:** Secondary Special Event Interrupt Enable bit<sup>(3)</sup>
  - 1 = Secondary Special Event Interrupt is enabled
  - 0 = Secondary Special Event Interrupt is disabled
- bit 10-7     **Unimplemented:** Read as '0'
- bit 6-4        **SCLKDIV<2:0>:** Secondary PWM Input Clock Prescaler<sup>(2)</sup>
  - 111 = Divide by 128, PWM resolution = (128/FSYSCLK)
  - 110 = Divide by 64, PWM resolution = (64/FSYSCLK)
  - 
  - 
  - 
  - 000 = Divide by 1, PWM resolution = 1/FSYSCLK (power-on default)
- bit 3-0        **SEVTPS<3:0>:** PWM Secondary Special Event Trigger Output Postscaler Select bits<sup>(2)</sup>
  - 1111 = 1:16 Postscale
  - 
  - 
  - 
  - 0001 = 1:2 Postscale
  - 0000 = 1:1 Postscale

- Note 1:** The SSESTAT bit is cleared by clearing the SSEIEN bit and corresponding bit in the IFSx register.
- 2:** These bits should be changed only when the PTEN bit (PTCON<15>) = 0.
- 3:** To clear the Secondary Special Event Interrupt, the user application must do the following:
- 1) First, clear the SSEIEN bit by setting it to '0'.
  - 2) Next, clear the Secondary Special Event Interrupt flag, IFS5<12>, by setting it to '0'.
  - 3) Finally, re-enable the STCON register by setting the SSEIEN bit equal to '1', if desired.
- The user application will not be able to clear the Secondary Special Event Interrupt flag as long as the SSEIEN bit is equal to '1'.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-8: SMTMR: SECONDARY MASTER TIME BASE TIMER REGISTER

| Bit Range | Bit<br>31/23/15/7          | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|----------------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0                        | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                          | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0                        | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                          | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R-0                        | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | SMTMR<15:8> <sup>(1)</sup> |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R-0                        | R-0               | R-0               | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | SMTMR<7:0> <sup>(1)</sup>  |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **SMTMR<15:0>:** Secondary Master Time Base Timer Value bits<sup>(1)</sup>

This timer increments with each PWM FSYCLK until the STPER value is reached.

**Note 1:** Min LSb = 1/FSYSCLK.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-10: PWMKEY: PWM UNLOCK REGISTER

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | W-0               | W-0               | W-0               | W-0               | W-0               | W-0               | W-0              | W-0              |
|           | PWMKEY<15:8>      |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | W-0               | W-0               | W-0               | W-0               | W-0               | W-0               | W-0              | W-0              |
|           | PWMKEY<7:0>       |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **PWMKEY<15:0>:** PWM Unlock bits

If the PWMLOCK Configuration bit is asserted (PWMLOCK = 0), the IOCONx registers are writable only after the proper sequence is written to the PWMKEY register. If the PWMLOCK Configuration bit is deasserted (PWMLOCK = 1), the IOCONx registers are writable at all times. For more information on the unlock sequence, refer to the **44.9 "Write Protection"** in **Section 44. Motor Control PWM (MCPWM)** (DS60001393) of the *"PIC32 Family Reference Manual"* for more information.

This register is implemented only in devices where the PWMLOCK Configuration bit is present in the DEVCFG3 Configuration register.

**Note:** The user must write two consecutive values of 0xABCD and 0x4321 to the PWMKEY register to perform an unlock operation if PWMLOCK = 0. Write access to any subsequent secure register must be the very next access following the unlock process. This is not an atomic operation and any CPU interrupts that occur during or immediately after an unlock sequence may cause writes to any PWM secure register to fail.





# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-11: PWMCONx: PWM CONTROL REGISTER 'x' ('x' = 1 THROUGH 12) (CONTINUED)

|           |  |
|-----------|--|
| bit 21    | <b>TRIGIEN:</b> Primary Trigger Interrupt Enable bit<br>1 = A primary trigger event generates an interrupt request<br>0 = A primary trigger event interrupts request is disabled   |
| bit 20    | <b>PWMLIEN:</b> PWM Low Phase Interrupt Enable bit<br>1 = When the PWM Timer is equal to 0x4, the PWMLIF flag = 1 and generates an interrupt request<br>0 = PWM Period event interrupt request is disabled   |
| bit 19    | <b>PWMHIEN:</b> PWM High Phase Interrupt Enable bit<br>1 = When the PWM Period matches the value in the PWM timer, an interrupt request is generated<br>0 = PWM Period event interrupt request is disabled, and the PWMHIF bit is cleared  |
| bit 18-16 | <b>Unimplemented:</b> Read as '0'  |
| bit 15    | <b>FLTSTAT:</b> Fault Interrupt Status bit <sup>(1)</sup><br>1 = Fault interrupt is pending<br>0 = No fault interrupt is pending<br>This bit is cleared by setting FLTIEN = 0.   |
| bit 14    | <b>CLTSTAT:</b> Current-Limit Interrupt Status bit <sup>(1)</sup><br>1 = Current-limit interrupt is pending<br>0 = No current-limit interrupt is pending<br>This bit is cleared by setting CLIEN = 0.  |
| bit 13-12 | <b>Unimplemented:</b> Read as '0'  |
| bit 11-10 | <b>ECAM&lt;1:0&gt;:</b> Edge/Center-Aligned Mode Enable bits <sup>(1)</sup><br>11 = Asymmetric Center-Aligned mode with simultaneous update (PWM(min) Duty Cycle Resolution = (1/FSYSCLK))<br>10 = Asymmetric Center-Aligned mode double update (PWM(min) Duty Cycle Resolution = (1/FSYSCLK))<br>01 = Symmetric Center-Aligned mode (PWM(min) Duty Cycle Resolution = (2/FSYSCLK))<br>00 = Edge-Aligned mode (PWM(min) Duty Cycle Resolution = (1/FSYSCLK)) |
| bit 9     | <b>ITB:</b> Independent Time Base Mode bit <sup>(2)</sup><br>1 = PHASEx registers provide time base period for this PWM generator<br>0 = PTPER/STPER register provides timing for this PWM generator based on the MTBS bit   |
| bit 8     | <b>Unimplemented:</b> Read as '0'  |
| bit 7-6   | <b>DTC&lt;1:0&gt;:</b> Dead Time Control bits<br>11 = Dead Time Compensation mode enabled<br>10 = Dead time function is disabled<br>01 = Negative dead time actively applied for Complementary Output mode <sup>(5)</sup><br>00 = Positive dead time actively applied for all output modes   |
| bit 5     | <b>DTCP:</b> Dead Time Compensation Polarity bit <sup>(5)</sup><br>1 = If the DTCMPx pin = 0, PWMxL is shortened, and PWMxH is lengthened<br>If the DTCMPx pin = 1, PWMxH is shortened, and PWMxL is lengthened<br>0 = If the DTCMPx pin = 0, PWMxH is shortened, and PWMxL is lengthened<br>If the DTCMPx pin = 1, PWMxL is shortened, and PWMxH is lengthened  |

- Note 1:** If PWM interrupts are enabled, software must clear the PWMCONx interrupt flags here first, followed second by the corresponding IFSx bit in the Interrupt controller. The corresponding PWM IFSx interrupt flag cannot be cleared if any of these local PWMCON interrupt bits are not cleared first. Failure to do so will result in an infinite interrupt loop.
- 2:** This bit should not be changed after the PWM is enabled (PTEN bit (PTCON<15>) = 1).
- 3:** To operate in External Period Reset mode, the ITB bit must be set to '1' and the CLMOD bit in the IOCONx register must be set to '0'.
- 4:** For Dead Time Compensation (DTCP) to be effective, DTC<1:0> must be set to '11'; otherwise, DTCP is ignored.
- 5:** Negative dead time is only implemented for Edge-Aligned mode.
- 6:** XPRES mode should only be used in Edge-Aligned mode with or without complimentary outputs. It does not support dead time compensation (i.e., duty cycle adjustment), which is selected when DTC<1:0> = 11.
- 7:** The clock source is one of the master time bases even if ITB = 1 is selected.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 29-11: PWMCONx: PWM CONTROL REGISTER 'x' ('x' = 1 THROUGH 12) (CONTINUED)

|       |  |
|-------|--|
| bit 4 | <b>PTDIR:</b> PWM Timer Direction bit <sup>(6)</sup><br>1 = PWM timer is decrementing<br>0 = PWM timer is incrementing   |
| bit 3 | <b>MTBS:</b> Master Time Base Select bit <sup>(7)</sup><br>1 = Secondary master time base is the clock source for the MCPWM module<br>0 = Primary master time base is the clock source for the MCPWM module  |
| bit 2 | <b>Unimplemented:</b> Read as '0'  |
| bit 1 | <b>XPRES:</b> External PWM Reset Control bit <sup>(3)</sup><br>1 = Current-limit source resets primary local time base for this PWM generator if it is in Independent Time Base mode and the PWM module enters the deassertion portion of the duty cycle<br>0 = External pins do not affect PWM time base<br><br><b>Note:</b> If the Current-Limit Reset signal is asserted during the active assertion time of the duty cycle, the time base will not Reset until two PWM clock cycles after the duty cycle transition from assertion to deassertion phase of the duty cycle. |
| bit 0 | <b>Unimplemented:</b> Read as '0'  |

- Note 1:** If PWM interrupts are enabled, software must clear the PWMCONx interrupt flags here first, followed second by the corresponding IFSx bit in the Interrupt controller. The corresponding PWM IFSx interrupt flag cannot be cleared if any of these local PWMCON interrupt bits are not cleared first. Failure to do so will result in an infinite interrupt loop.
- 2:** This bit should not be changed after the PWM is enabled (PTEN bit (PTCON<15>) = 1).
- 3:** To operate in External Period Reset mode, the ITB bit must be set to '1' and the CLMOD bit in the IOCONx register must be set to '0'.
- 4:** For Dead Time Compensation (DTCP) to be effective, DTC<1:0> must be set to '11'; otherwise, DTCP is ignored.
- 5:** Negative dead time is only implemented for Edge-Aligned mode.
- 6:** XPRES mode should only be used in Edge-Aligned mode with or without complimentary outputs. It does not support dead time compensation (i.e., duty cycle adjustment), which is selected when DTC<1:0> = 11.
- 7:** The clock source is one of the master time bases even if ITB = 1 is selected.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

bit 29-26 **CLSRC<3:0>**: Current-Limit Control Signal Source select bit for PWM Generator 'x'(2,4)

These bits specify the current-limit control signal source.

1111 = FLT15  
1110 = Reserved  
1101 = Reserved  
1100 = Comparator 5  
1011 = Comparator 4  
1010 = Comparator 3  
1001 = Comparator 2  
1000 = Comparator 1  
0111 = FLT8  
0110 = FLT7  
0101 = FLT6  
0100 = FLT5  
0011 = FLT4  
0010 = FLT3  
0001 = FLT2  
0000 = FLT1

bit 25 **CLPOL**: Current-Limit Polarity bits for PWM Generator 'x'(2,4)

1 = The selected current-limit source is active-low  
0 = The selected current-limit source is active-high

- Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).
- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLT<sub>x</sub> inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLT<sub>x</sub> pin. In addition, DTCMP functions are fixed to specific FLT<sub>x</sub> inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1:** (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)

```
PWMCON1bits.DTC = 0b11; //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1; //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0110; //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1; //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111; //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1:** (DTCMP1 = Current Limit = Fault = FLT3 pin)

```
PWMCON1bits.DTC = 0b11; //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1; //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010; //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1; //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010; //Enable Fault for PWM1 on FLT3 pin
```

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

- bit 24     **CLMOD:** Current-Limit Mode Enable bit for PWM Generator 'x'<sup>(2,4)</sup>  
          1 = Current-limit function is enabled  
          0 = Current-limit function is disabled, current-limit overrides disabled (current-limit interrupts can still be generated). If Faults are enabled, FLTMOD will override the CLMOD bit.  
          Changes take effect on the next PWM cycle boundary following PWM being enabled, and subsequently on each PWM cycle boundary. When updating CLMOD from '1' to '0', if the current-limit input is still active, the current-limit override condition will not be removed.
- bit 23     **Unimplemented:** Read as '0'

- Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).
- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLT<sub>x</sub> inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLT<sub>x</sub> pin. In addition, DTCMP functions are fixed to specific FLT<sub>x</sub> inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1: (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)**

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b01110;       //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111;      //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1: (DTCMP1 = Current Limit = Fault = FLT3 pin)**

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010;       //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010;      //Enable Fault for PWM1 on FLT3 pin
```

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

bit 22-19 **FLTSRC<3:0>**: Fault Control Signal Source Select bits for PWM Generator 'x'<sup>(2,4)</sup>

These bits specify the Fault control source.

1111 = FLT15  
1110 = Reserved  
1101 = Reserved  
1100 = Comparator 5  
1011 = Comparator 4  
1010 = Comparator 3  
1001 = Comparator 2  
1000 = Comparator 1  
0111 = FLT8  
0110 = FLT7  
0101 = FLT6  
0100 = FLT5  
0011 = FLT4  
0010 = FLT3  
0001 = FLT2  
0000 = FLT1

bit 18 **FLTPOL**: Fault Polarity bits for PWM Generator 'x'<sup>(2)</sup>

1 = The selected fault source is active-low  
0 = The selected fault source is active-high

- Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).
- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLT<sub>x</sub> inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLT<sub>x</sub> pin. In addition, DTCMP functions are fixed to specific FLT<sub>x</sub> inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1:** (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0110;       //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111;      //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1:** (DTCMP1 = Current Limit = Fault = FLT3 pin)

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010;       //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010;      //Enable Fault for PWM1 on FLT3 pin
```

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

- bit 17-16 **FLTMOD<1:0>**: Fault Mode bits for PWM Generator 'x'<sup>(4)</sup>
- 11 = Fault input is disabled, no fault overrides possible. (fault interrupts can still be generated)
  - 10 = Reserved
  - 01 = Selected fault source forces PWMxH, PWMxL pins to FLTDAT<1:0> values (cycle by cycle)
  - 00 = Selected fault source forces PWMxH, PWMxL pins to FLTDAT<1:0> values (Latched condition)
- Changes take effect on the next PWM cycle boundary following PWM being enabled, and subsequently on each PWM cycle boundary. When updating FLTMOD<1:0> from '00' or '01' to '11' (disabled), if the fault input is still active the fault override condition will not be removed. If enabled, Faults will override the CLMOD bit setting.
- bit 15 **PENH**: PWMxH Output Pin Ownership bit<sup>(1)</sup>
- 1 = PWM module controls PWMxH pin
  - 0 = GPIO module controls PWMxH pin
- bit 14 **PENL**: PWMxL Output Pin Ownership bit<sup>(1)</sup>
- 1 = PWM module controls PWMxL pin
  - 0 = GPIO module controls PWMxL pin
- bit 13 **POLH**: PWMxH Output Pin Polarity bit<sup>(2)</sup>
- 1 = PWMxH pin is active-low
  - 0 = PWMxH pin is active-high

**Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).

- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLT<sub>x</sub> inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLT<sub>x</sub> pin. In addition, DTCMP functions are fixed to specific FLT<sub>x</sub> inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1: (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)**

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0110;        //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111;       //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1: (DTCMP1 = Current Limit = Fault = FLT3 pin)**

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010;        //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1;           //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010;       //Enable Fault for PWM1 on FLT3 pin
```



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

|           |  |
|-----------|--|
| bit 12    | <b>POLL:</b> PWMxL Output Pin Polarity bit <sup>(2)</sup><br>1 = PWMxL pin is active-low<br>0 = PWMxL pin is active-high   |
| bit 11-10 | <b>PMOD&lt;1:0&gt;:</b> PWM 'x' I/O Pin Mode bits <sup>(2)</sup><br>11 = PWMxL output is held at logic '0' (adjusted by the POLL bit)<br>10 = PWM I/O pin pair is in Push-Pull Output mode<br>01 = PWM I/O pin pair is in Redundant Output mode<br>00 = PWM I/O pin pair is in Complementary Output mode |
| bit 9     | <b>OVRENH:</b> Override Enable for PWMxH Pin bit<br>1 = OVRDAT<1> provides data for output on PWMxH pin<br>0 = PWM generator provides data for PWMxH pin   |
| bit 8     | <b>OVRENL:</b> Override Enable for PWMxL Pin bit<br>1 = OVRDAT<0> provides data for output on PWMxL pin<br>0 = PWM generator provides data for PWMxL pin   |
| bit 7-6   | <b>OVRDAT&lt;1:0&gt;:</b> State <sup>(3)</sup> for PWMxH, PWMxL Pins if Override is Enabled bits<br>If OVRENH = 1, OVRDAT<1> provides data for PWMxH<br>If OVRENL = 1, OVRDAT<0> provides data for PWMxL   |

- Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).
- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLT<sub>x</sub> inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLT<sub>x</sub> pin. In addition, DTCMP functions are fixed to specific FLT<sub>x</sub> inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1:** (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)

```
PWMCON1bits.DTC = 0b11; //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1; //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0110; //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1; //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111; //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1:** (DTCMP1 = Current Limit = Fault = FLT3 pin)

```
PWMCON1bits.DTC = 0b11; //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1; //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010; //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1; //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010; //Enable Fault for PWM1 on FLT3 pin
```

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-12: IOCONx: PWMX I/O CONTROL REGISTER 'x' ('x' = 1 THROUGH 9) (CONTINUED)

- bit 5-4 **FLTDAT<1:0>**: State<sup>(3)</sup> for PWMxH and PWMxL Pins if FLTMOD is Enabled bits<sup>(2)</sup>  
If FLTMOD<1:0> (IOCONx<17:16>) = 00 or 01, one of the following Fault modes is enabled:  
If fault is active, FLTDAT<1> provides the state for PWMxH  
If fault is active, FLTDAT<0> provides the state for PWMxL  
If fault is inactive, FLTDAT<1:0> bits are ignored
- bit 3-2 **CLDAT<1:0>**: State for PWMxH and PWMxL Pins if CLMOD is Enabled bits<sup>(3)</sup>  
If CLMOD (IOCONx<24>) = 1, Current-Limit mode is enabled, as follows:  
If current limit is active, CLTDAT<1> provides the state for PWMxH  
If current limit is active, CLTDAT<0> provides the state for PWMxL  
If current limit is inactive, CLTDAT<1:0> bits are ignored
- bit 1 **SWAP**: SWAP PWMxH and PWMxL Pins bit  
1 = PWMxH output signal is connected to PWMxL pin; PWMxL output signal is connected to PWMxH pin  
0 = PWMxH and PWMxL output signals pins are mapped to their respective pins
- bit 0 **OSYNC**: Output Override Synchronization bit  
1 = Output overrides through the OVRDAT<1:0> bits are synchronized to the PWM time base  
0 = Output overrides through the OVRDAT<1:0> bits occur on next CPU clock boundary

- Note 1:** During PWM initialization, if the PWMLOCK fuse bit is 'enabled' (logic '0'), the control on the state of the PWMxL/PWMxH output pins rests solely with the PENH and PENL bits. However, these bits are at '0', which leaves the pin control with the I/O module. Care must be taken to not inadvertently set the TRIS bits to output, which could impose an incorrect output on the PWMxH/PWMxL pins even if there are external pull-up and pull-down resistors. The data direction for the pins must be set to input if tri-state behavior is desired or be driven to the appropriate logic states. The PENH and PENL bits must always be initialized prior to enabling the MCPWM module (PTEN bit = 1).
- 2:** These bits must not be changed after the MCPWM module is enabled (PTEN bit = 1).
- 3:** State represents Active/Inactive state of the PWM, depending on the POLH and POLL bits. For example, if FLTDAT<1> is set to '1' and POLH is set to '1', the PWMxH pin will be at logic level 0 (active level) when a Fault occurs.
- 4:** If (PWMLOCK = 0), these bits are writable only after the proper sequence is written to the PWMKEY register. If (PWMLOCK = 1), these bits are writable at all times. The user application must write two consecutive values of (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation for the IOCONx register if PWMLOCK = 1. Write access to a IOCONx register must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. This is not an atomic operation, and therefore, any CPU interrupts that occur during or immediately after an unlock sequence may cause the IOCONx SFR write access to fail.

**Note:** Dead Time Compensation, Current-Limit, and Faults share common inputs on the FLTx inputs ('x' = 1-8, and 15). Therefore, it is not recommended that a user application assign these multiple functions on the same Fault FLTx pin. In addition, DTCMP functions are fixed to specific FLTx inputs, where Current-Limit, (CLSRC<3:0> bits) and Faults (FLTSRC<3:0> bits) can be assigned to any one of 15 unique and separate inputs. For example, if a user application was required to assign multiple simultaneous Fault, Current-Limit, DTCMP to a single PWM1. Refer to the following examples for both desirable and undesirable practices.

**Desirable Example PWM1:** (DTCMP1 = FLT3 pin, Current Limit = FLT7 pin, Fault = FLT8 pin)

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0110;        //Enable current limit for PWM1 on FLT7 pin
IOCON1bits.FLTMOD = 1;            //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0111;       //Enable Fault for PWM1 on FLT8 pin
```

**Undesirable Example: PWM1:** (DTCMP1 = Current Limit = Fault = FLT3 pin)

```
PWMCON1bits.DTC = 0b11;           //Enable DTCMP1 input on FLT3 function pin
IOCON1bits.CLMOD = 1;             //Enable PWM1 Current-Limit mode
IOCON1bits.CLSRC = 0b0010;        //Enable current limit for PWM1 on FLT3 pin
IOCON1bits.FLTMOD = 1;            //Enable PWM1 Fault mode
IOCON1bits.FLTSRC = 0b0010;       //Enable Fault for PWM1 on FLT3 pin
```

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-13: PDCx: PWM GENERATOR DUTY CYCLE REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | PDC<15:8>         |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | PDC<7:0>          |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **PDC<15:0>:** Primary PWM Generator 'x' Duty Cycle Value bits<sup>(2)</sup>

If Edge-Aligned mode is enabled (ECAM<1:0> bits (PWMCONx<11:10>) = 00), these bits specify the trailing edge instance of the ON time and controls the duty cycle directly (PWM Resolution = (1/FSYCLK)).

If one of the Center-Aligned mode is enabled (ECAM<1:0> (PWMCONx<11:10>) = 01, 10, or 11), these bits specify the compare instance for 'leading edge' level transition (PWM Resolution = (2/FSYCLK)).

**Note 1:** In Independent PWM mode, PMOD<1:0> (IOCONx<11:10>) = 11, the PDCx register controls the PWMxH duty cycle only. In Complementary, Redundant and Push-Pull PWM modes (PMOD<1:0> = 00, 01, or 10), the PDCx register controls the duty cycle of both the PWMxH and PWMxL.

**2:**  $PDCx = ((FSYSCLK / (F_{PWM} * PTCN<PCLKDIV>)) * \text{Desired Duty Cycle})$   
 $F_{PWM}$  = User Desired PWM Frequency

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-14: SDCx: PWM SECONDARY DUTY CYCLE REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | SDC<15:8>         |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | SDC<7:0>          |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **SDC<15:0>:** Secondary Duty Cycle bits for PWMx output pin

If Edge-Aligned mode is enabled (ECAM<1:0> (PWMCONx<11:10>) = 00) these bits are unused.

If Symmetric Center-Aligned mode is enabled (ECAM<1:0> (PWMCONx<11:10>) = 01), these bits are updated transparently to the user. Loads to the PDCx register automatically copy over to the SDCx register.

If Asymmetric Center-Aligned mode is enabled (ECAM<1:0> (PWMCONx<11:10>) = 10 or 11), these bits specify the compare instance for 'trailing edge' level transition (PWM Resolution = (2/FSYCLK)).

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-15: PHASE<sub>x</sub>: PWM PRIMARY PHASE SHIFT REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | PHASE<15:8>       |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | PHASE<7:0>        |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **PHASE<15:0>**: PWM Phase Shift Value or Independent Time Base Period bits for the PWM Generator bits<sup>(6)</sup>

Phase shifting is used to offset the start of a PWM Generator's time base period, relative to a master time base, as well as the generated duty cycle. Also, the effects on the operation of the PWM signals through any external control signals, such as current-limit, Fault, and dead time compensation, are also shifted in time.

**Note 1:** If the ITB bit (PWMCONx<9>) = 0, the following applies based on the mode of operation:

Complementary, Redundant and Push-Pull Output modes (PMOD<1:0> (IOCONx<11:10>) = 00, 01, or 10) PHASE<15:0> = Phase shift value for PWMxH and PWMxL outputs

**2:** If the ITB bit = 1, the following applies based on the mode of operation:

Complementary, Redundant, and Push-Pull Output modes (PMOD<1:0> = 00, 01, or 10) PHASE<15:0> = local time base period value for TMRx

**3:** A Phase offset that exceeds the PWM period will lead to unpredictable results.

**4:** The minimum period value is 0x0008.

**5:** The SDCx register is used in Independent PWM mode only (PMOD<1:0> = 11). When used in Independent PWM mode, the SDCx register controls the PWMxL duty cycle.

**6:** PHASE<sub>x</sub> = (FSYSCLK / (F<sub>PWM</sub> \* PTCN<PCLKDIV>))

F<sub>PWM</sub> = User Desired PWM Frequency

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-16: DTRx: PWM DEAD TIME REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | DTR<13:8>         |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | DTR<7:0>          |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **DTR<13:0>:** Unsigned 14-bit Dead Time Value for PWMxH Dead Time Unit bits

These bits specify the leading edge dead time count between the PWMxH and PWMxL. The time base for the count is the same as for the PWM generator.

The dead time period is typically set equal to the switching times of the power transistors in the application circuits. It is specifically intended for use in Complementary Output mode. The use of dead time in any other mode may generate unexpected or unpredictable results. If the duty cycle value in the DC register equals '0', or is greater than or equal to the Period, dead time compensation is ignored. The values for Duty Cycle + Dead Time + Dead Time Compensation must not exceed the value for the Period register minus 1. If the sum exceeds the Period Register minus 1, unexpected results may occur. The values for Duty Cycle + Dead Time - Dead Time Compensation must be greater than '0', or unexpected results may occur.

**Note:** DTR<13:0> and ALTDTR<13:0> must be ≥ 6 while using Leading Edge Blanking.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-17: ALTDTR<sub>x</sub>: PWM ALTERNATE DEAD TIME REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | ALTDTR<13:8>      |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | ALTDTR<7:0>       |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **ALTDTR<13:0>:** Unsigned 14-bit Dead Time Value for PWM<sub>x</sub>L Dead Time Unit bits

These bits specify the trailing edge dead time count between the PWM<sub>x</sub>H and PWM<sub>x</sub>L. The time base for the count is the same as for the PWM generator.

The alternate dead time period is typically set equal to the switching times of the power transistors in the application circuits. It is specifically intended for use in Complementary Output mode. The use of dead time in any other mode may generate unexpected or unpredictable results. If the duty cycle value in the DC register equals '0', or is greater than or equal to the Period, alternate dead time compensation is ignored. The values for Duty Cycle + Dead Time + ALT Dead Time Compensation must not exceed the value for the Period Register minus 1. If the sum exceeds the Period Register -minus1, unexpected results may occur. The values for Duty Cycle + Dead Time minus Alternate Dead Time Compensation must be greater than '0', or unexpected results may occur.

**Note:** DTR<13:0> and ALTDTR<13:0> must be ≥ 6 while using Leading Edge Blanking.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-18: DTCOMP<sub>x</sub>: DEAD TIME COMPENSATION REGISTER 'x' ('x' = 1 THROUGH 12)

| Bit Range | Bit<br>31/23/15/7          | Bit<br>30/22/14/6 | Bit<br>29/21/13/5           | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|----------------------------|-------------------|-----------------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0                        | U-0               | U-0                         | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                          | —                 | —                           | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0                        | U-0               | U-0                         | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                          | —                 | —                           | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0                        | U-0               | R/W-0                       | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                          | —                 | COMP<13:8> <sup>(1,2)</sup> |                   |                   |                   |                  |                  |
| 7:0       | R-0                        | R-0               | R-0                         | R-0               | R-0               | R-0               | R-0              | R-0              |
|           | COMP<7:0> <sup>(1,2)</sup> |                   |                             |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit                                      W = Writable bit                                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                                      '1' = Bit is set                                      '0' = Bit is cleared                                      x = Bit is unknown

bit 31-16    **Unimplemented:** Read as '0'

bit 15-0    **COMP<13:0>:** Dead Time Compensation Value bits<sup>(1,2)</sup>

Dead time compensation value if Dead Time compensation mode is enabled.

**Note 1:** COMP<13:0> Min LSb = 1 / FSYSCLK for ECAM<1:0> bits (PWMCON<sub>x</sub><11:10>) = `\0b00` Edge-Aligned mode; COMP<13:0> Min LSb = 2 / FSYSCLK for ECAM<1:0> bits (PWMCON<sub>x</sub><11:10>) = `\0b00` Center-Aligned mode.

**2:** When Dead Time compensation mode is selected through the DTC<1:0> bits in the PWMCON<sub>x</sub> register, an external pin, CMP<sub>x</sub> (i.e., FLT<sub>x</sub>) connected to the Dead Time Compensation module input signals, cause the value in the COMP<sub>x</sub> register to be added to or subtracted from the PWM<sub>x</sub> duty cycle. The dead time compensation input signals are sampled at the end of a PWM cycle for use in the next PWM cycle. The modification of the duty cycle duration through the CMP<sub>x</sub> registers occurs during the end (trailing edge) of the duty cycle. Dead time compensation is available only for Positive Dead Time mode. The CMP<sub>x</sub> value must be less than one-half the value of the duty cycle register, PDC<sub>x</sub>; otherwise, unpredictable behavior will result. Dead time compensation will not apply for a duty cycle of zero. In this case, the PWM output will remain zero regardless of the state of the CMP<sub>x</sub> input pin.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-19: TRIGx: PWM PRIMARY TRIGGER COMPARE VALUE REGISTER 'x' ('x' = 1 THROUGH 12)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | TRGCMP<15:8>      |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | TRGCMP<7:0>       |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **TRGCMP<15:0>:** Trigger Compare Value bits

These bits specify the value to match against the local time base register PTMR to generate a trigger to the ADC module and an interrupt if the TRGIEN bit (PWMCONx<21>) is set.

**Note:** To trigger at the period boundary, set TRIGx to 0x0 and not the PTPER period value.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-20: TRGCONx: PWM TRIGGER CONTROL REGISTER 'x' ('x' = 1 THROUGH 12)

| Bit Range | Bit<br>31/23/15/7    | Bit<br>30/22/14/6     | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3          | Bit<br>26/18/10/2 | Bit<br>25/17/9/1            | Bit<br>24/16/8/0 |
|-----------|----------------------|-----------------------|-------------------|-------------------|----------------------------|-------------------|-----------------------------|------------------|
| 31:24     | U-0                  | U-0                   | U-0               | U-0               | U-0                        | U-0               | U-0                         | U-0              |
|           | —                    | —                     | —                 | —                 | —                          | —                 | —                           | —                |
| 23:16     | U-0                  | U-0                   | U-0               | U-0               | U-0                        | U-0               | U-0                         | U-0              |
|           | —                    | —                     | —                 | —                 | —                          | —                 | —                           | —                |
| 15:8      | R/W-0                | R/W-0                 | R/W-0             | R/W-0             | R/W-0                      | R/W-0             | R/W-0                       | R/W-0            |
|           | TRGDIV<3:0>          |                       |                   |                   | TRGSEL<1:0> <sup>(1)</sup> |                   | STRGSEL<1:0> <sup>(1)</sup> |                  |
| 7:0       | R/W-0                | R/W-0                 | U-0               | U-0               | U-0                        | U-0               | U-0                         | U-0              |
|           | DTM <sup>(1,2)</sup> | STRGIS <sup>(1)</sup> | —                 | —                 | —                          | —                 | —                           | —                |

### Legend:

R = Readable bit                                      W = Writable bit                                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                                      '1' = Bit is set                                      '0' = Bit is cleared                                      x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-12 **TRGDIV<3:0>:** Trigger 'x' Output Divider bits  
 1111 = Trigger output for every sixteenth trigger event  
 •  
 •  
 •

0010 = Trigger output for every third trigger event  
 0001 = Trigger output for every second trigger event  
 0000 = Trigger output for every trigger event

bit 11-10 **TRGSEL<1:0>:** Trigger Cycle Selection for Dual Cycle PWM Cycles (Center-Aligned and Push-Pull)<sup>(1)</sup>  
 This bit field has no effect on the raw trigger generation for single cycle PWM modes such as edge-aligned PWM. Each time a raw comparison event occurs, the raw event is processed by the trigger divider.  
 11 = Reserved, default to same behavior as TRGSEL<1:0> = 00.  
 10 = When a trigger comparison match event occurs in the incrementing phase in the dual cycle PWM mode (PTDIR = 0), a trigger event output is generated if the trigger divider has counted the appropriate number of trigger events.  
 01 = When a trigger comparison match event occurs in the decrementing phase in the dual cycle PWM mode (PTDIR = 1), a trigger event output is generated if the trigger divider has counted the appropriate number of trigger events.  
 00 = When a trigger comparison match event occurs, generate a trigger event output if the trigger divider has counted the appropriate number of raw trigger events. For dual cycle PWM modes such as Center-Aligned mode and Push-Pull mode, the raw trigger event is generated twice every cycle. However, TRIGx/STRIGx compare values of '0' or equal to the PERIOD match register will only generate one interrupt even in the dual cycle modes.

- Note 1:** These bits must not be changed after the MCPWM module is enabled (PTEN bit (PTCON<15>) = 1).  
**Note 2:** The secondary trigger event is generated regardless of the setting of the DTM bit.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-20: TRGCONx: PWM TRIGGER CONTROL REGISTER 'x' ('x' = 1 THROUGH 12) (CONTINUED)

- bit 9-8     **STRGSEL<1:0>**: Secondary Trigger Cycle Selection bits for Dual Cycle PWM Cycles (Center-Aligned and Push-Pull)<sup>(1)</sup>
- These bits have no effect on the raw secondary PWM trigger generation for single cycle PWM modes such as edge aligned PWM. Each time a raw comparison event occurs, the raw event is processed by the secondary PWM trigger divider.
- 11 = Reserved, default to same behavior as STRGSEL<1:0> = 00
  - 10 = When a secondary PWM trigger comparison match event occurs in the second half of a dual cycle PWM mode (PTDIR = 0), generate a secondary PWM trigger event output if the secondary PWM trigger divider has counted the appropriate number of secondary PWM trigger events.
  - 01 = When a secondary PWM trigger comparison match event occurs in the first half of a dual cycle PWM mode (PTDIR = 1), generate a trigger event output if the secondary PWM trigger divider has counted the appropriate number of secondary PWM trigger events.
  - 00 = When a secondary PWM trigger comparison match event occurs, generate a secondary PWM trigger event output if the trigger divider has counted the appropriate number of raw secondary PWM trigger events. For two cycle PWM modes such as Center-Aligned mode and Push-Pull mode, the raw secondary PWM trigger event is generated twice.
- bit 7       **DTM**: Dual ADC Trigger Mode<sup>(1, 2)</sup>
- 1 = Secondary trigger event is combined with the primary trigger event for purposes of creating a combined ADC trigger
  - 0 = Secondary trigger event is not combined with the primary trigger event for purposes of creating a combined ADC trigger
- bit 6       **STRGIS**: Secondary Trigger Interrupt Select<sup>(1)</sup>
- This bit should be changed by the user only when PTEN = 0.
- 1 = Selects the Secondary Trigger Register (STRIGx) based events for interrupts
  - 0 = When the DTM bit (TRGCONx<7>) is clear (= 0), TRIGx-based events for interrupts are selected. When the DTM bit is set (= 1), the logical OR of both STRIGx and TRIGx based triggers for interrupts are selected.
- bit 5-0     **Unimplemented**: Read as '0'

- Note 1:** These bits must not be changed after the MCPWM module is enabled (PTEN bit (PTCON<15>) = 1).
- 2:** The secondary trigger event is generated regardless of the setting of the DTM bit.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-23: LEBCONx: LEADING-EDGE BLANKING CONTROL REGISTER 'x' (‘x’ = 1 THROUGH 12)

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | U-0              | U-0              |
|           | PHR               | PHF               | PLR               | PLF               | FLTLEBEN          | CLLEBEN           | —                | —                |
| 7:0       | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as ‘0’

-n = Value at POR

‘1’ = Bit is set

‘0’ = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as ‘0’

bit 15 **PHR:** PWMxH Rising Edge Trigger Enable bit

1 = Rising edge of PWMxH will trigger/retrigger the Leading-Edge Blanking counter

0 = Rising edge of PWMxH will not trigger/retrigger the Leading-Edge Blanking counter

bit 14 **PHF:** PWMxH Falling Edge Trigger Enable bit

1 = Falling edge of PWMxH will trigger/retrigger the Leading-Edge Blanking counter

0 = Falling edge of PWMxH will not trigger/retrigger the Leading-Edge Blanking counter

bit 13 **PLR:** PWMxL Rising Edge Trigger Enable bit

1 = Rising edge of PWMxL will trigger/retrigger the Leading-Edge Blanking counter

0 = Rising edge of PWMxL will not trigger/retrigger the Leading-Edge Blanking counter

bit 12 **PLF:** PWMxL Falling Edge Trigger Enable bit

1 = Falling edge of PWMxL will trigger/retrigger the Leading-Edge Blanking counter

0 = Falling edge of PWMxL will not trigger/retrigger the Leading-Edge Blanking counter

bit 11 **FLTLEBEN:** Fault Input Leading-Edge Blanking Enable bit

1 = Leading-Edge Blanking is applied to selected fault input

0 = Leading-Edge Blanking is not applied to selected fault input

bit 10 **CLLEBEN:** Current-Limit Leading-Edge Blanking Enable bit

1 = Leading-Edge Blanking is applied to selected current-limit input

0 = Leading-Edge Blanking is not applied to selected current-limit input

bit 9-0 **Unimplemented:** Read as ‘0’

**Note:** DTR<13:0> and ALTDTR<13:0> must be ≥ 6 while using Leading Edge Blanking.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-24: LEBDLY<sub>x</sub>: LEADING-EDGE BLANKING DELAY REGISTER 'x' (‘x’ = 1 THROUGH 12)

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | U-0               | U-0               | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | —                 | —                 | LEB<11:8>         |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | LEB<7:0>          |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
‘1’ = Bit is set

U = Unimplemented bit, read as ‘0’  
‘0’ = Bit is cleared  
x = Bit is unknown

bit 31-12 **Unimplemented:** Read as ‘0’

bit 11-0 **LEB<11:0>:** Leading-Edge Blanking Delay bits for Current-Limit and Fault Inputs bits

These bits specify the time period for which the selected current limit and fault signals are blanked or delayed following the selected edge transition of the PWM signals. This retriggerable counter has the PWM module clock source (SYSCLK) as the time base.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 29-25: AUXCONx: PWM AUXILIARY CONTROL REGISTER 'x' ('x' = 1 THROUGH 9)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5           | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-----------------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0                         | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                           | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0                         | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                           | —                 | —                 | —                 | —                | —                |
| 15:8      | U-0               | U-0               | U-0                         | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                           | —                 | —                 | —                 | —                | —                |
| 7:0       | U-0               | U-0               | R/W-0                       | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | —                 | —                 | CHOPSEL<3:0> <sup>(1)</sup> |                   |                   |                   | CHOPHEN          | CHOPLEN          |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-6      **Unimplemented:** Read as '0'

bit 5-2      **CHOPSEL<3:0>:** PWM Chop Clock Source Select bits<sup>(1)</sup>

The selected signal will enable and disable (CHOP) the selected PWM outputs.

1111 = Reserved.

•  
•

1010 = Reserved.

1001 = PWM9H selected as CHOP clock source (68-pin variant only)

1000 = PWM8H selected as CHOP clock source (68-pin variant only)

0111 = PWM7H selected as CHOP clock source (68-pin variant only)

0110 = PWM6H selected as CHOP clock source

•  
•

0001 = PWM1H selected as CHOP clock source

0000 = Chop clock generator selected as CHOP clock source

bit 1      **CHOPHEN:** PWMxH Output Chopping Enable bit

1 = PWMxH chopping function is enabled

0 = PWMxH chopping function is disabled

bit 0      **CHOPLEN:** PWMxL Output Chopping Enable bit

1 = PWMxL chopping function is enabled

0 = PWMxL chopping function is disabled

**Note 1:** This bit should be changed only when the PTEN bit (PTCON<15>) = 0.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 29-26: PTMRx: PWM TIMER REGISTER 'x' ('x' = 1 THROUGH 12)

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 23:16     | U-0               | U-0               | U-0               | U-0               | U-0               | U-0               | U-0              | U-0              |
|           | —                 | —                 | —                 | —                 | —                 | —                 | —                | —                |
| 15:8      | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | TMR<15:8>         |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R/W-0             | R/W-0             | R/W-1             | R/W-0             | R/W-0             | R/W-0             | R/W-0            | R/W-0            |
|           | TMR<7:0>          |                   |                   |                   |                   |                   |                  |                  |

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15-0 **TMR<15:0>:** PWM Timer bits

When the ECAM<1:0> bits (PWMCONx<11:10>) = 00, the counter counts upwards until a period match forces rollover.

When the ECAM<1:0> bits (PWMCONx<11:10>) ≠ 00, the counter counts downwards starting with a master time base synchronization signal to 0 and then counts upwards until the next synchronization.



# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

## 30.0 HIGH/LOW-VOLTAGE DETECT (HLVD)

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 38. “High/Low-Voltage Detect (HLVD)”** (DS60001408), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

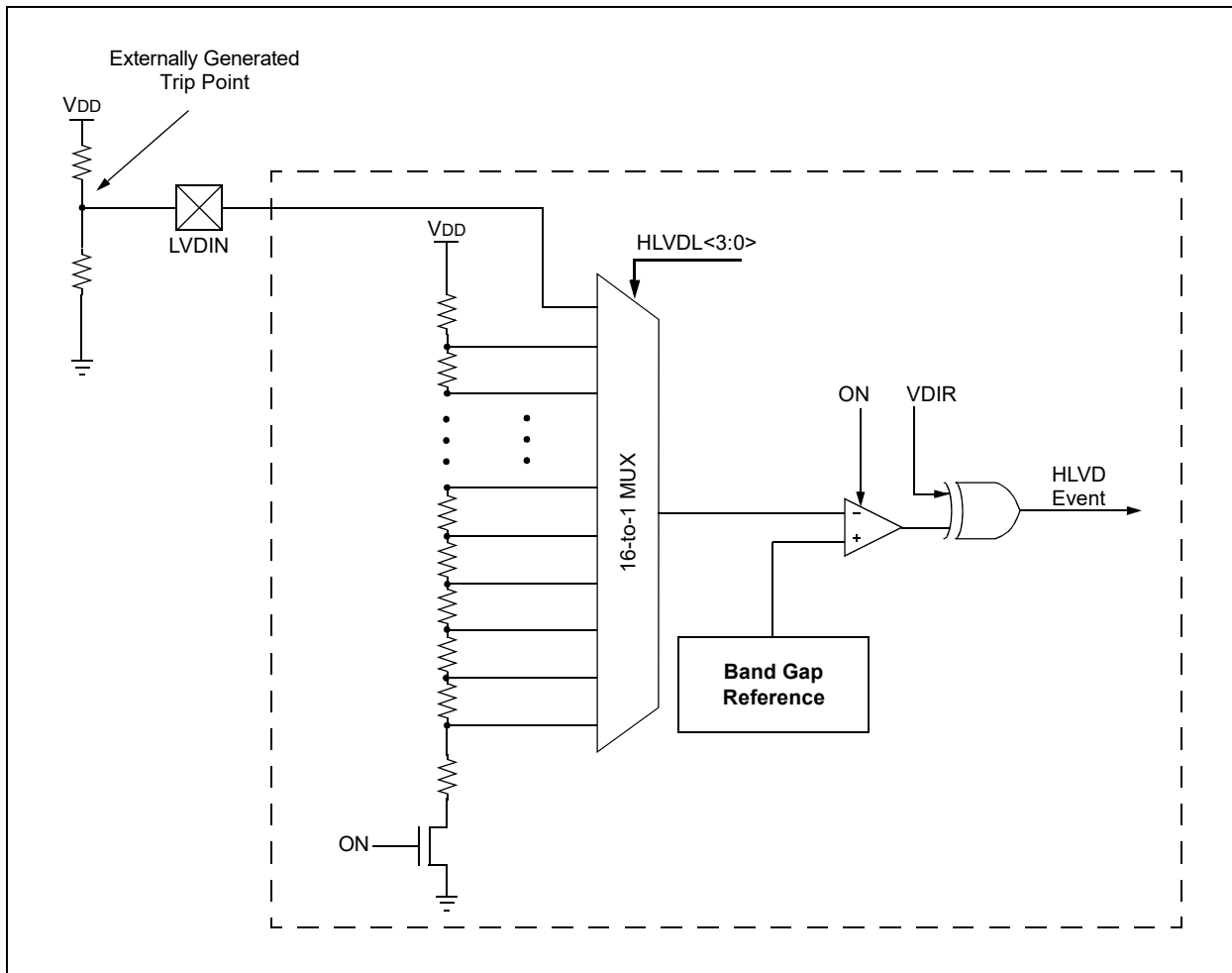
The High/Low-Voltage Detect (HLVD) module is a programmable circuit that can be used to specify both the device voltage trip point and the direction of change. When enabled, a HLVD event will act to disable the Flash controller from executing a programming sequence. This module is used to ensure the supply voltage is sufficient for programming.

The HLVD module is an interrupt-driven supply-level detection. The voltage detection monitors the internal power supply.

The HLVD module provides the following key features:

- Detection hysteresis
- Detection of low-to-high or high-to-low voltage changes
- Generation of Non-Maskable Interrupts (NMI)
- LVDIN pin to provide external voltage trip point

**FIGURE 30-1: PROGRAMMABLE HLVD MODULE BLOCK DIAGRAM**



30.1 Control Registers

TABLE 30-1: HLVD REGISTER MAP

| Virtual Address<br>(BF80_#) | Register<br>Name <sup>(1)</sup> | Bit Range | Bits  |       |       |       |       |       |      |        |        |      |      |      |            |      | All Resets |      |
|-----------------------------|---------------------------------|-----------|-------|-------|-------|-------|-------|-------|------|--------|--------|------|------|------|------------|------|------------|------|
|                             |                                 |           | 31/15 | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8   | 23/7   | 22/6 | 21/5 | 20/4 | 19/3       | 18/2 |            | 17/1 |
| 1800                        | HLVDCON                         | 31:16     | —     | —     | —     | —     | —     | —     | —    | —      | —      | —    | —    | —    | —          | —    | —          | 0000 |
|                             |                                 | 15:0      | ON    | —     | —     | —     | VDIR  | BGVST | —    | HLVDET | DISOUT | —    | —    | —    | HLVDL<3:0> |      |            | 0000 |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**Note 1:** The register in this table has corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 30-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

| Bit Range                 | Bit 31/23/15/7  | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3               | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0          |
|---------------------------|-----------------|----------------|----------------|----------------|------------------------------|----------------|---------------|------------------------|
| 31:24                     | U-0<br>—        | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—                     | U-0<br>—       | U-0<br>—      | U-0<br>—               |
| 23:16                     | U-0<br>—        | U-0<br>—       | U-0<br>—       | U-0<br>—       | U-0<br>—                     | U-0<br>—       | U-0<br>—      | U-0<br>—               |
| 15:8                      | R/W-0<br>ON     | U-0<br>—       | U-0<br>—       | U-0<br>—       | R/W-0<br>VDIR <sup>(1)</sup> | R/W-0<br>BGVST | U-1<br>—      | HS/HC, R/W-0<br>HLVDET |
| 7:0                       | R/S-0<br>DISOUT | U-0<br>—       | U-0<br>—       | U-0<br>—       | R/W-0                        | R/W-0          | R/W-0         | R/W-0                  |
| HLVDL<3:0> <sup>(1)</sup> |                 |                |                |                |                              |                |               |                        |

|                   |                   |                                    |                    |
|-------------------|-------------------|------------------------------------|--------------------|
| <b>Legend:</b>    | HS = Hardware Set | HC = Hardware Clear                | S = Settable bit   |
| R = Readable bit  | W = Writable bit  | U = Unimplemented bit, read as '0' |                    |
| -n = Value at POR | '1' = Bit is set  | '0' = Bit is cleared               | x = Bit is unknown |

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** HLVD Module Enable bit  
 1 = HLVD module is enabled  
 0 = HLVD module is disabled

bit 14-12 **Unimplemented:** Read as '0'

bit 11 **VDIR:** Voltage Change Direction Select bit<sup>(1)</sup>  
 1 = Event occurs when voltage equals or exceeds trip point (HLVDL<3:0>)  
 0 = Event occurs when voltage equals or falls below trip point (HLVDL<3:0>)

bit 10 **BGVST:** Band Gap Reference Voltages Stable Status bit  
 1 = Indicates internal band gap voltage references is stable  
 0 = Indicates internal band gap voltage reference is not stable

**Note:** This bit is readable when the HLVD module is disabled (ON bit is equal to '0').

bit 9 **Unimplemented:** Read as '1'

bit 8 **HLVDET:** High/Low-Voltage Detection Event Status bit  
 1 = Indicates HLVD Event interrupt is active  
 0 = Indicates HLVD Event interrupt is not active

bit 7 **DISOUT:** Disable HLVD Output bit  
 1 = Enable output of HLVD Comparator. Once set, can only be cleared by setting the ON bit to '0'.  
 0 = Disable output of HLVD Comparator (default Reset value). Will be forced to '0' whenever the ON bit is equal to '0'.

bit 6-4 **Unimplemented:** Read as '0'

**Note 1:** To avoid false HLVD events, all HLVD module setting changes should occur only when the module is disabled (ON bit is equal to '0').

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 30-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

bit 3-0 **HLVDL<3:0>**: High/Low-Voltage Detection Limit Select bits<sup>(1)</sup>

1111 = 1.2V (Selects external LVDIN pin)

1110 = Reserved, Do not use

1101 = Reserved, Do not use

1100 = Reserved, Do not use

1011 = Reserved, Do not use

1010 = 2.4V

1001 = 2.5V

1000 = 2.697V

0111 = 2.791V

0110 = 3.0V

0101 = 3.288V

0100 = 3.529V

0011 = Reserved; do not use

0010 = Reserved; do not use

0001 = Reserved; do not use

0000 = Reserved; do not use

**Note 1:** To avoid false HLVD events, all HLVD module setting changes should occur only when the module is disabled (ON bit is equal to '0').

# PIC32MK GPG/MCJ with CAN FD Family

## 31.0 POWER-SAVING FEATURES

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 10. “Power-Saving Features”** (DS60001130), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

This section describes the power-saving features on the PIC32MK GPG/MCJ CAN FD family of devices. These devices have multiple power domains and offer various methods and modes that allow the user to balance the power consumption with device performance.

### 31.1 Power Saving with CPU Running

When the CPU is running, power consumption can be controlled by reducing the CPU clock frequency or selecting a lower power clock source (i.e., LPRC or Sosc).

In addition, the Peripheral Bus Scaling mode is available for each peripheral bus where peripherals are clocked at reduced speed by selecting a higher divider for the associated PBCLKx, or by disabling the clock completely.

### 31.2 Power-Saving with CPU Halted

Peripherals and the CPU can be Halted or disabled to further reduce power consumption.

#### 31.2.1 SLEEP MODE

Sleep mode has the lowest power consumption of the device power-saving operating modes. The CPU and most peripherals are Halted and the associated clocks are disabled. Select peripherals can continue to operate in Sleep mode and can be used to wake the device from Sleep. See the individual peripheral module sections for descriptions of behavior in Sleep mode.

Sleep mode includes the following characteristics:

- There can be a wake-up delay based on the oscillator selection
- The Fail-Safe Clock Monitor (FSCM) does not operate during Sleep mode
- The BOR circuit remains operative during Sleep mode
- The WDT, if enabled, is not automatically cleared prior to entering Sleep mode

- Some peripherals can continue to operate at limited functionality in Sleep mode. These peripherals include I/O pins that detect a change in the input signal, WDT, ADC, UART and peripherals that use an external clock input or the internal LPRC oscillator (e.g., RTCC, Timer1 and Input Capture).
- I/O pins continue to sink or source current in the same manner as they do when the device is not in Sleep

The processor will exit, or ‘wake-up’, from Sleep on one of the following events:

- On any interrupt from an enabled source that is operating in Sleep. The interrupt priority must be greater than the current CPU priority.
- On any form of device Reset
- On a WDT time-out

If the interrupt priority is lower than or equal to the current priority, the CPU will remain Halted, but the peripheral bus clocks will start running and the device will enter into Idle mode.

#### 31.2.2 IDLE MODE

In Idle mode, the CPU is Halted; however, all clocks are still enabled. This allows peripherals to continue to operate. Peripherals can be individually configured to Halt when entering Idle by setting their respective SIDL bit. Latency, when exiting Idle mode, is very low due to the CPU oscillator source remaining active.

The device enters Idle mode when the SLPEN bit (OSCCON<4>) is clear and a WAIT instruction is executed.

The processor will wake or exit from Idle mode on the following events:

- On any interrupt event for which the interrupt source is enabled. The priority of the interrupt event must be greater than the current priority of the CPU. If the priority of the interrupt event is lower than or equal to current priority of the CPU, the CPU will remain Halted and the device will remain in Idle mode.
- On any form of device Reset
- On a WDT time-out interrupt

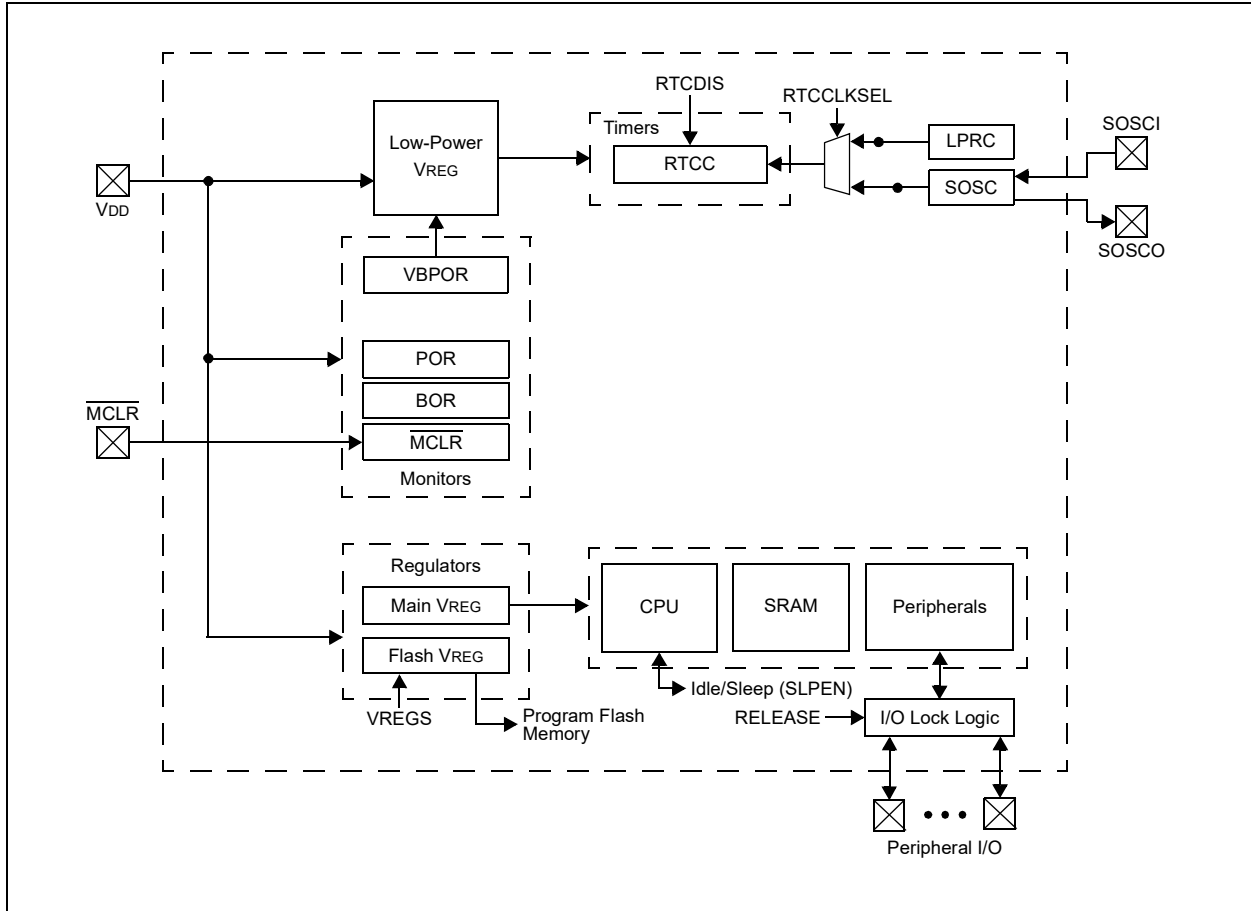
#### 31.2.3 POWER-SAVING MODES

Figure 31-1 shows a block diagram and the related power-saving features. The various blocks are controlled by the following Configuration bit settings and SFRs:

- RTCCCLKSEL (RTCCON <9:8>)
- SLPEN (OSCCON<4>)
- VREGS (PWRCON<0>)

# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 31-1: LOW-POWER DEVICE BLOCK DIAGRAM



# PIC32MK GPG/MCJ with CAN FD Family

---

## 31.3 Peripheral Module Disable

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled using the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral are also disabled, so writes to those registers do not have effect and read values are invalid. If the associated peripheral PMD bit was previously = 1, subsequently clearing the bit will require the user to reinitialize the peripheral. The only exception to this is the DMA module.

To disable a peripheral, the associated PMDx bit must be set to '1'. To enable a peripheral, the associated PMDx bit must be cleared (default). See [Table 31-1](#) for more information.

|   |
|---|
| <p><b>Note:</b> Disabling a peripheral module while its ON bit is set, may result in undefined behavior. The ON bit for the associated peripheral module must be cleared prior to disable a module via the PMDx bits.</p> |
|---|



**TABLE 31-1: PERIPHERAL MODULE DISABLE REGISTER SUMMARY**

| Virtual Address<br>(BF80_#) | Register<br>Name      | Bit Range | Bits  |       |       |                       |        |        |                       |                       |                       |                       |                       |                       |                       |                       |                       | All Resets <sup>(1)</sup> |       |
|-----------------------------|-----------------------|-----------|-------|-------|-------|-----------------------|--------|--------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|---------------------------|-------|
|                             |                       |           | 31/15 | 30/14 | 29/13 | 28/12                 | 27/11  | 26/10  | 25/9                  | 24/8                  | 23/7                  | 22/6                  | 21/5                  | 20/4                  | 19/3                  | 18/2                  | 17/1                  |                           | 16/0  |
| 0040                        | PMD1 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | CLC4MD | CLC3MD | CLC2MD                | CLC1MD                | —                     | —                     | —                     | HLVDMD                | —                     | —                     | —                     | —                         | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | —                     | CTMUMD                | —                     | DAC2MD                | DAC1MD                | —                     | —                     | —                     | —                     | —                         | ADCMD |
| 0050                        | PMD2 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | —      | —      | —                     | —                     | —                     | —                     | —                     | OPA5MD                | —                     | OPA3MD                | OPA2MD                | OPA1MD                    | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | —                     | —                     | —                     | —                     | —                     | CMP5MD                | C4MPMD                | C3MPMD                | CMP2MD                | CMP1MD                    | 0000  |
| 0060                        | PMD3 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | —      | —      | —                     | OC9MD                 | OC8MD                 | OC7MD                 | OC6MD                 | OC5MD                 | OC4MD                 | OC3MD                 | OC2MD                 | OC1MD                     | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | —                     | IC9MD                 | IC8MD                 | IC7MD                 | IC6MD                 | IC5MD                 | IC4MD                 | IC3MD                 | IC2MD                 | IC1MD                     | 0000  |
| 0070                        | PMD4 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | —      | —      | —                     | PWM9MD <sup>(4)</sup> | PWM8MD <sup>(4)</sup> | PWM7MD <sup>(4)</sup> | PWM6MD <sup>(4)</sup> | PWM5MD <sup>(4)</sup> | PWM4MD <sup>(4)</sup> | PWM3MD <sup>(4)</sup> | PWM2MD <sup>(4)</sup> | PWM1MD <sup>(4)</sup>     | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | —                     | T9MD                  | T8MD                  | T7MD                  | T6MD                  | T5MD                  | T4MD                  | T3MD                  | T2MD                  | T1MD                      | 0000  |
| 0080                        | PMD5 <sup>(1,2)</sup> | 31:16     | —     | —     | —     | CAN1MD <sup>(3)</sup> | —      | —      | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | I2C2MD                | I2C1MD                    | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | SPI2MD                | SPI1MD                | —                     | —                     | —                     | —                     | —                     | —                     | U2MD                  | U1MD                      | 0000  |
| 0090                        | PMD6 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | —      | —      | QEI3MD <sup>(4)</sup> | QEI2MD <sup>(4)</sup> | QEI1MD <sup>(4)</sup> | —                     | —                     | —                     | —                     | —                     | —                     | —                         | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | REFO4MD               | REFO3MD               | REFO2MD               | REFO1MD               | —                     | —                     | —                     | —                     | —                     | —                         | —     |
| 00A0                        | PMD7 <sup>(2)</sup>   | 31:16     | —     | —     | —     | —                     | —      | —      | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                     | —                         | 0000  |
|                             |                       | 15:0      | —     | —     | —     | —                     | —      | —      | —                     | —                     | —                     | —                     | —                     | —                     | DMAMD                 | —                     | —                     | —                         | —     |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: Reset values are dependent on the device variant.
  - 2: For any associated PMDx bit, '0' = clocks enabled to the peripheral; '1' = For associated peripheral, clocks are disabled, SFRs are reset, and CPU read/write is invalid.
  - 3: Not available in "GPG" variants.
  - 4: Only available in "MC" variants.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 31-2: PERIPHERAL MODULE DISABLE BITS AND LOCATIONS**

| Peripheral       | PMDx Bit Name <sup>(1)</sup> | Register Name and Bit Location |
|------------------|------------------------------|--------------------------------|
| ADC1-ADC7        | ADC1MD                       | PMD1<0>                        |
| CDAC1            | DAC1MD                       | PMD1<5>                        |
| CDAC2            | DAC2MD                       | PMD1<6>                        |
| CTMU             | CTMU1MD                      | PMD1<8>                        |
| HLVD             | HLVDM                        | PMD1<20>                       |
| CLC1             | CLC1MD                       | PMD1<24>                       |
| CLC2             | CLC2MD                       | PMD1<25>                       |
| CLC3             | CLC3MD                       | PMD1<26>                       |
| CLC4             | CLC4MD                       | PMD1<27>                       |
| Comparator 1     | C1MD                         | PMD2<0>                        |
| Comparator 2     | C2MD                         | PMD2<1>                        |
| Comparator 3     | C3MD                         | PMD2<2>                        |
| Comparator 4     | C4MD                         | PMD2<3>                        |
| Comparator 5     | C5MD                         | PMD2<4>                        |
| Op amp 1         | OPA1MD                       | PMD2<16>                       |
| Op amp 2         | OPA2MD                       | PMD2<17>                       |
| Op amp 3         | OPA3MD                       | PMD2<18>                       |
| Op amp 5         | OPA5MD                       | PMD2<20>                       |
| Input Capture 1  | IC1MD                        | PMD3<0>                        |
| Input Capture 2  | IC2MD                        | PMD3<1>                        |
| Input Capture 3  | IC3MD                        | PMD3<2>                        |
| Input Capture 4  | IC4MD                        | PMD3<3>                        |
| Input Capture 5  | IC5MD                        | PMD3<4>                        |
| Input Capture 6  | IC6MD                        | PMD3<5>                        |
| Input Capture 7  | IC7MD                        | PMD3<6>                        |
| Input Capture 8  | IC8MD                        | PMD3<7>                        |
| Input Capture 9  | IC9MD                        | PMD3<8>                        |
| Output Compare 1 | OC1MD                        | PMD3<16>                       |
| Output Compare 2 | OC2MD                        | PMD3<17>                       |
| Output Compare 3 | OC3MD                        | PMD3<18>                       |
| Output Compare 4 | OC4MD                        | PMD3<19>                       |
| Output Compare 5 | OC5MD                        | PMD3<20>                       |
| Output Compare 6 | OC6MD                        | PMD3<21>                       |
| Output Compare 7 | OC7MD                        | PMD3<22>                       |
| Output Compare 8 | OC8MD                        | PMD3<23>                       |
| Output Compare 9 | OC9MD                        | PMD3<24>                       |
| Timer1           | T1MD                         | PMD4<0>                        |
| Timer2           | T2MD                         | PMD4<1>                        |
| Timer3           | T3MD                         | PMD4<2>                        |

**Note 1:** For any associated PMDx bit, 0 = clocks enabled to the peripheral; 1 = For associated peripheral, clocks are disabled, SFRs are reset, and CPU read/write is invalid. If the associated peripheral PMD bit was previously = 1, subsequently clearing the bit will require the user to reinitialize the peripheral. The only exception to this is the DMA module.

**2:** Not available in “GPG” variants.

**3:** Only available in “MC” variants.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 31-2: PERIPHERAL MODULE DISABLE BITS AND LOCATIONS (CONTINUED)**

| Peripheral          | PMDx Bit Name <sup>(1)</sup> | Register Name and Bit Location |
|---------------------|------------------------------|--------------------------------|
| Timer4              | T4MD                         | PMD4<3>                        |
| Timer5              | T5MD                         | PMD4<4>                        |
| Timer6              | T6MD                         | PMD4<5>                        |
| Timer7              | T7MD                         | PMD4<6>                        |
| Timer8              | T8MD                         | PMD4<7>                        |
| Timer9              | T9MD                         | PMD4<8>                        |
| PWM1 <sup>(3)</sup> | PWM1MD                       | PMD4<16>                       |
| PWM2 <sup>(3)</sup> | PWM2MD                       | PMD4<17>                       |
| PWM3 <sup>(3)</sup> | PWM3MD                       | PMD4<18>                       |
| PWM4 <sup>(3)</sup> | PWM4MD                       | PMD4<19>                       |
| PWM5 <sup>(3)</sup> | PWM5MD                       | PMD4<20>                       |
| PWM6 <sup>(3)</sup> | PWM6MD                       | PMD4<21>                       |
| PWM7 <sup>(3)</sup> | PWM7MD                       | PMD4<22>                       |
| PWM8 <sup>(3)</sup> | PWM8MD                       | PMD4<23>                       |
| PWM9 <sup>(3)</sup> | PWM9MD                       | PMD4<24>                       |
| Uart1               | U1MD                         | PMD5<0>                        |
| Uart2               | U2MD                         | PMD5<1>                        |
| SPI1                | SPI1MD                       | PMD5<8>                        |
| SPI2                | SPI2MD                       | PMD5<9>                        |
| I2C1                | I2C1MD                       | PMD<16>                        |
| I2C2                | I2C2MD                       | PMD<17>                        |
| CAN1 <sup>(2)</sup> | CAN1MD                       | PMD5<28>                       |
| Reference Clock 1   | REFO1MD                      | PMD6<8>                        |
| Reference Clock 2   | REFO2MD                      | PMD6<9>                        |
| Reference Clock 3   | REFO3MD                      | PMD6<10>                       |
| Reference Clock 4   | REFO4MD                      | PMD6<11>                       |
| QE1 <sup>(3)</sup>  | QE1MD                        | PMD6<24>                       |
| QE2 <sup>(3)</sup>  | QE2MD                        | PMD6<25>                       |
| QE3 <sup>(3)</sup>  | QE3MD                        | PMD6<26>                       |
| DMA                 | DMAMD                        | PMD7<4>                        |

**Note 1:** For any associated PMDx bit, 0 = clocks enabled to the peripheral; 1 = For associated peripheral, clocks are disabled, SFRs are reset, and CPU read/write is invalid. If the associated peripheral PMD bit was previously = 1, subsequently clearing the bit will require the user to reinitialize the peripheral. The only exception to this is the DMA module.

**2:** Not available in “GPG” variants.

**3:** Only available in “MC” variants.

# PIC32MK GPG/MCJ with CAN FD Family

---

## 31.3.1 CONTROLLING CONFIGURATION CHANGES

Because peripherals can be disabled during run time, some restrictions on disabling peripherals are needed to prevent accidental configuration changes. PIC32MK GPG/MCJ with CAN FD Family of devices include two features to prevent alterations to enabled or disabled peripherals:

- Control Register Lock Sequence
- Configuration Bit Select Lock

### 31.3.1.1 Control Register Lock

Under normal operation, writes to the PMDx registers are not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the PMDLOCK Configuration bit (CFGCON<12>). Setting the PMDLOCK bit prevents writes to the control registers and clearing the PMDLOCK bit allows writes.

To set or clear the PMDLOCK bit, an unlock sequence must be executed. Refer to **Section 42. “Oscillators with Enhanced PLL”** (DS60001250) in the *“PIC32 Family Reference Manual”* for details.

### 31.3.1.2 Configuration Bit Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the PMDx registers. The PMDL1WAY Configuration bit (DEVCFG3<28>) blocks the PMDLOCK bit from being cleared after it has been set once. If the PMDLOCK bit remains set, the register unlock procedure does not execute, and the PPS control registers cannot be written to. The only way to clear the bit and re-enable PMD functionality is to perform a device Reset.

# PIC32MK GPG/MCJ with CAN FD Family

## 32.0 SPECIAL FEATURES

**Note:** This data sheet summarizes the features of the PIC32MK GPG/MCJ with CAN FD Family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 32. “Configuration”** (DS60001124) and **Section 33. “Programming and Diagnostics”** (DS60001129), which are available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site ([www.microchip.com/pic32](http://www.microchip.com/pic32)).

PIC32MK GPG/MCJ with CAN FD Family of devices include several features intended to maximize application flexibility and reliability and minimize cost through elimination of external components. These are:

- Flexible device configuration
- Joint Test Action Group (JTAG) interface
- In-Circuit Serial Programming™ (ICSP™)
- Internal temperature sensor

### 32.1 Configuration Bits

PIC32MK GPG/MCJ with CAN FD Family of devices contain two Boot Flash memories (Boot Flash 1 and Boot Flash 2), each with an associated configuration space. These configuration spaces can be programmed to contain various device configurations. Configuration space that is aliased by the Lower Boot Alias memory region is used to provide values for Configuration registers listed below. See **3.1.1 “Boot Flash Sequence and Configuration Spaces”** for more information.

- [DEVSIGN0: Device Signature Word 0 Register](#)
- [DEVCP0: Device Code-Protect 0 Register](#)
- [DEVCFG0/ADEVCFG0: Device/Alternate Device Configuration Word 0](#)
- [DEVCFG1/ADEVCFG1: Device/Alternate Device Configuration Word 1](#)
- [DEVCFG2/ADEVCFG2: Device/Alternate Device Configuration Word 2](#)
- [DEVCFG3/ADEVCFG3: Device/Alternate Device Configuration Word 3](#)

The following run-time programmable Configuration registers provide additional configuration control:

- [CFGCON: Configuration Control Register](#)
- [CFGPG: Permission Group Configuration Register](#)

In addition, the DEVID register ([Register 32-9](#)) provides device and revision information, the DEVADC1 through DEVADC5 registers ([Register 32-10](#)) provide ADC module calibration data, and the DEVSNO and DEVSNO3 registers contain a unique serial number of the device ([Register 32-11](#)).

**Note:** Do not use Word program operation (NVMOP<3:0> = 0001) when programming the device Words that are described in this section.

32.2 Registers

TABLE 32-1: DEVCFG: DEVICE CONFIGURATION WORD SUMMARY

| Virtual Address (BFC0 #) | Register Name | Bit Range | Bits         |               |         |          |         |                |                 |              |                |         |             |               |               |               |            | All Resets    |      |      |
|--------------------------|---------------|-----------|--------------|---------------|---------|----------|---------|----------------|-----------------|--------------|----------------|---------|-------------|---------------|---------------|---------------|------------|---------------|------|------|
|                          |               |           | 31/15        | 30/14         | 29/13   | 28/12    | 27/11   | 26/10          | 25/9            | 24/8         | 23/7           | 22/6    | 21/5        | 20/4          | 19/3          | 18/2          | 17/1       |               | 16/0 |      |
| 3FC0                     | DEVCFG3       | 31:16     | —            | —             | IOL1WAY | PMDL1WAY | PGL1WAY | —              | —               | —            | —              | —       | —           | PWMLOCK       | —             | —             | —          | —             | xxxx |      |
|                          |               | 15:0      | USERID<15:0> |               |         |          |         |                |                 |              |                |         |             |               |               |               |            | xxxx          |      |      |
| 3FC4                     | DEVCFG2       | 31:16     | —            | —             | BORSEL  | FDSEN    | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | FPLLIDIV<2:0> | xxxx |      |
|                          |               | 15:0      | —            | FPLLMULT<6:0> |         |          |         |                |                 | FPLLICK      | FPLL RNG<2:0>  |         |             | —             | FPLLIDIV<2:0> |               |            | xxxx          |      |      |
| 3FC8                     | DEVCFG1       | 31:16     | FDMTEN       | DMTCNT<4:0>   |         |          |         | FWDTWINSZ<1:0> |                 | FWDTEN       | WINDIS         | WDTSPGM | WDTPSR<4:0> |               |               |               | xxxx       |               |      |      |
|                          |               | 15:0      | FCKSM<1:0>   |               | —       | —        | —       | —              | OSCI0FNC        | POSCMOD<1:0> |                | IES0    | LPOSCEN     | DMTINTV<2:0>  |               | FNOSC<2:0>    |            | xxxx          |      |      |
| 3FCC                     | DEVCFG0       | 31:16     | —            | EJTAGBEN      | —       | —        | POSCAGC | —              | POSCAGCDLY<1:0> |              | POSCFGAIN<1:0> |         | POSCBOOST   | POSCGAIN<1:0> |               | SOSCGAIN<2:0> |            | xxxx          |      |      |
|                          |               | 15:0      | SMCLR        | DBGPER<2:0>   |         |          | —       | FSLEEP         | FECCCON<1:0>    |              | —              | BOOTISA | TRCEN       | ICESEL<1:0>   |               | JTAGEN        | DEBUG<1:0> |               | xxxx |      |
| 3FD0                     | DEVCP3        | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | xxxx |      |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
| 3FD4                     | DEVCP2        | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | xxxx |      |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
| 3FD8                     | DEVCP1        | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | xxxx |      |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
| 3FDC                     | DEVCP0        | 31:16     | —            | —             | —       | CP       | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | xxxx |      |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
| 3FF0                     | DEVSEQ 3      | 31:16     | CSEQ<15:0>   |               |         |          |         |                |                 |              |                |         |             |               |               |               |            | xxxx          |      |      |
|                          |               | 15:0      | TSEQ<15:0>   |               |         |          |         |                |                 |              |                |         |             |               |               |               |            | xxxx          |      |      |
| 3FF4                     | DEVSEQ 2      | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | —    |
| 3FF8                     | DEVSEQ 1      | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | —    |
| 3FFC                     | DEVSEQ0       | 31:16     | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | xxxx |
|                          |               | 15:0      | —            | —             | —       | —        | —       | —              | —               | —            | —              | —       | —           | —             | —             | —             | —          | —             | —    | —    |

Legend: x = unknown value on Reset; See register description detail for more information. Reset values are shown in hexadecimal.

**TABLE 32-2: ADEVCFG: ALTERNATE DEVICE CONFIGURATION WORD SUMMARY**

| Virtual Address (BFC0 #) | Register Name | Bit Range | Bits          |             |         |          |         |                |              |                 |                |           |               |               |            |            |      | All Resets    |      |      |      |
|--------------------------|---------------|-----------|---------------|-------------|---------|----------|---------|----------------|--------------|-----------------|----------------|-----------|---------------|---------------|------------|------------|------|---------------|------|------|------|
|                          |               |           | 31/15         | 30/14       | 29/13   | 28/12    | 27/11   | 26/10          | 25/9         | 24/8            | 23/7           | 22/6      | 21/5          | 20/4          | 19/3       | 18/2       | 17/1 |               | 16/0 |      |      |
| 3F40                     | ADEVCFG3      | 31:16     | —             | —           | IOL1WAY | PMDL1WAY | PGL1WAY | —              | —            | —               | —              | —         | —             | PWMLOCK       | —          | —          | —    | —             | xxxx |      |      |
|                          |               | 15:0      | USERID<15:0>  |             |         |          |         |                |              |                 |                |           |               |               |            |            |      | xxxx          |      |      |      |
| 3F44                     | ADEVCFG2      | 31:16     | —             | —           | BORSEL  | FDMEN    | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | FPLLODIV<2:0> | xxxx |      |      |
|                          |               | 15:0      | FPLLMULT<6:0> |             |         |          |         |                |              |                 |                |           |               |               |            |            |      | xxxx          |      |      |      |
| 3F48                     | ADEVCFG1      | 31:16     | FDMTEN        | DMTCNT<4:0> |         |          |         | FWDTWINSZ<1:0> |              | FWDTEN          | WINDIS         | WDTSPGM   | WDTPSR<4:0>   |               |            |            | xxxx |               |      |      |      |
|                          |               | 15:0      | FCKSM<1:0>    |             | —       | —        | —       | —              | OSCIOfNC     | POSCMOD<1:0>    | IES0           | LPOSCEN   | DMTINTV<2:0>  |               | FNOSC<2:0> |            | xxxx |               |      |      |      |
| 3F4C                     | ADEVCFG0      | 31:16     | —             | EJTABEN     | —       | —        | —       | POSCAGC        | —            | POSCAGCDLY<1:0> | POSCFGAIN<1:0> | POSCBOOST | POSCGAIN<1:0> | SOSCGAIN<2:0> |            |            | xxxx |               |      |      |      |
|                          |               | 15:0      | SMCLR         | DBGPER<2:0> |         |          | —       | FSLEEP         | FECCCON<1:0> |                 | —              | BOOTISA   | TRCEN         | ICESEL<1:0>   | JTAGEN     | DEBUG<1:0> |      | xxxx          |      |      |      |
| 3F50                     | ADEVCP3       | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | xxxx |      |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
| 3F54                     | ADEVCP2       | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | xxxx |      |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
| 3F58                     | ADEVCP1       | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | xxxx |      |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
| 3F5C                     | ADEVCP0       | 31:16     | —             | —           | —       | —        | CP      | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | xxxx |      |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
| 3F70                     | ADEVSEQ 3     | 31:16     | CSEQ<15:0>    |             |         |          |         |                |              |                 |                |           |               |               |            |            |      | xxxx          |      |      |      |
|                          |               | 15:0      | TSEQ<15:0>    |             |         |          |         |                |              |                 |                |           |               |               |            |            |      | xxxx          |      |      |      |
| 3F74                     | ADEVSEQ 2     | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | —    |
| 3F78                     | ADEVSEQ 1     | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | —    |
| 3F7C                     | ADEVSEQ0      | 31:16     | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | xxxx |
|                          |               | 15:0      | —             | —           | —       | —        | —       | —              | —            | —               | —              | —         | —             | —             | —          | —          | —    | —             | —    | —    | —    |

**Legend:** x = unknown value on Reset; See register description detail for more information. Reset values are shown in hexadecimal.

TABLE 32-3: DEVICE ID, REVISION, AND CONFIGURATION SUMMARY

| Virtual Address<br>(BF80_#) | Register<br>Name | Bit Range | Bits         |       |             |         |              |            |           |      |      |      |            |              |            |       | All Resets <sup>(2)</sup> |        |      |      |      |
|-----------------------------|------------------|-----------|--------------|-------|-------------|---------|--------------|------------|-----------|------|------|------|------------|--------------|------------|-------|---------------------------|--------|------|------|------|
|                             |                  |           | 31/15        | 30/14 | 29/13       | 28/12   | 27/11        | 26/10      | 25/9      | 24/8 | 23/7 | 22/6 | 21/5       | 20/4         | 19/3       | 18/2  |                           | 17/1   | 16/0 |      |      |
| 0000                        | CFGCON           | 31:16     | —            | —     | —           | —       | —            | ADCPRI     | —         | —    | —    | —    | —          | —            | —          | —     | ICACLK                    | OCACLK | 0000 |      |      |
|                             |                  | 15:0      | —            | —     | IOLOCK      | PMDLOCK | PGLOCK       | —          | —         | —    | —    | —    | —          | ECCCON<1:0>  | JTAGEN     | TROEN | —                         | TDOEN  | 00xx |      |      |
| 0020                        | DEVID            | 31:16     | VER<3:0>     |       |             |         | DEVID<27:16> |            |           |      |      |      |            |              |            |       |                           |        |      |      | xxxx |
|                             |                  | 15:0      | DEVID<15:0>  |       |             |         |              |            |           |      |      |      |            |              |            |       |                           |        |      | xxxx |      |
| 0030                        | SYSKEY           | 31:16     | SYSKEY<31:0> |       |             |         |              |            |           |      |      |      |            |              |            |       |                           |        |      | 0000 |      |
|                             |                  | 15:0      | SYSKEY<31:0> |       |             |         |              |            |           |      |      |      |            |              |            |       |                           |        |      | 0000 |      |
| 00E0                        | CFGPG            | 31:16     | —            | —     | JTAGPG<1:0> | —       | —            | ADCPG<1:0> | FCPG<1:0> |      | —    | —    | —          | —            | —          | —     | —                         | —      | 0000 |      |      |
|                             |                  | 15:0      | —            | —     | CAN1PG<1:0> | —       | —            | —          | —         | —    | —    | —    | DMAPG<1:0> | CPUDSPG<1:0> | CPUPG<1:0> | —     | —                         | —      | 0000 |      |      |

**Legend:** x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note** 1: This register has corresponding CLR, SET, and INV registers at its virtual address, plus an offset of 0x4, 0x8, and 0xC, respectively. See [10.2 “CLR, SET, and INV Registers”](#) for more information.
- 2: Reset values are dependent on the device variant.
- 3: This register is not available on 64-pin devices.



**TABLE 32-4: DEVICE ADC CALIBRATION SUMMARY**

| Virtual Address<br>(BFC4_#) | Register<br>Name       | Bit Range | Bits                         |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets <sup>(1)</sup> |
|-----------------------------|------------------------|-----------|------------------------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|---------------------------|
|                             |                        |           | 31/15                        | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |                           |
| 5000                        | DEVADC0 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5004                        | DEVADC1 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5008                        | DEVADC2 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 500C                        | DEVADC3 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5010                        | DEVADC4 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5014                        | DEVADC5 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5018                        | DEVADC7 <sup>(2)</sup> | 31:16     | ADC Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                        | 15:0      | ADC Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |

**Legend:** x = unknown value on Reset.

**Note 1:** Reset values are dependent on the device variant.

**Note 2:** Before enabling the ADC, the user application must initialize the ADC calibration codes by copying them from the factory programmed DEVADCx Flash locations into the ADCxCFG special function registers, respectively.

**TABLE 32-5: DEVICE EE DATA CALIBRATION SUMMARY**

| Virtual Address<br>(BFC4_#) | Register<br>Name | Bit Range | Bits                             |       |       |       |       |       |      |      |      |      |      |      |      |      | All Resets <sup>(1)</sup> |
|-----------------------------|------------------|-----------|----------------------------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|---------------------------|
|                             |                  |           | 31/15                            | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 |                           |
| 5030                        | DEVEE0           | 31:16     | EE Data Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | EE Data Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5034                        | DEVEE1           | 31:16     | EE Data Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | EE Data Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 5038                        | DEVEE2           | 31:16     | EE Data Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | EE Data Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
| 503C                        | DEVEE3           | 31:16     | EE Data Calibration Data <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | EE Data Calibration Data <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      | xxxx                      |

**Legend:** x = unknown value on Reset.

**Note 1:** Reset values are device dependent. This is a device by device unique permanent number programmed into the device by the factory before shipment.

**TABLE 32-6: DEVICE SERIAL NUMBER SUMMARY**

| Virtual Address<br>(BFC4_#) | Register<br>Name | Bit Range | Bits                         |       |       |       |       |       |      |      |      |      |      |      |      |      |      | All Resets <sup>(1)</sup> |
|-----------------------------|------------------|-----------|------------------------------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|---------------------------|
|                             |                  |           | 31/15                        | 30/14 | 29/13 | 28/12 | 27/11 | 26/10 | 25/9 | 24/8 | 23/7 | 22/6 | 21/5 | 20/4 | 19/3 | 18/2 | 17/1 |                           |
| 5020                        | DEVSN0           | 31:16     | Device Serial Number <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | Device Serial Number <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
| 5024                        | DEVSN1           | 31:16     | Device Serial Number <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | Device Serial Number <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
| 5028                        | DEVSN2           | 31:16     | Device Serial Number <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | Device Serial Number <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
| 502C                        | DEVSN3           | 31:16     | Device Serial Number <31:16> |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |
|                             |                  | 15:0      | Device Serial Number <15:0>  |       |       |       |       |       |      |      |      |      |      |      |      |      |      | xxxx                      |

**Legend:** x = unknown value on Reset.

**Note 1:** Reset values are dependent on the device variant.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 32-1: DEVSIGN0: DEVICE SIGNATURE WORD 0 REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-0            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 23:16     | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

|                   |                  |                      |                                    |
|-------------------|------------------|----------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit | W = Writable bit     | U = Unimplemented bit, read as '0' |
| R = Readable bit  | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown                 |
| -n = Value at POR |                  |                      |                                    |

bit 31 **Reserved:** Write as '0'

bit 30-0 **Reserved:** Write as '1'

**REGISTER 32-2: DEVCP0: DEVICE CODE-PROTECT 0 REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-1            | r-1            | r-1            | R/P            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | CP             | —              | —              | —             | —             |
| 23:16     | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 15:8      | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |
| 7:0       | r-1            | r-1            | r-1            | r-1            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | —              | —              | —              | —             | —             |

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit | P = Programmable bit               |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31-29 **Reserved:** Write as '1'

bit 28 **CP:** Code-Protect bit

Prevents boot and program Flash memory from being read or modified by an external programming device.

1 = Protection is disabled

0 = Protection is enabled

bit 27-0 **Reserved:** Write as '1'

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 32-3: DEVCFG0/ADEVCFG0: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 0

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1   | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|-----------------|---------------|
| 31:24     | r-x            | R/P            | r-1            | r-1            | R/P            | r              | R/P             | R/P           |
|           | —              | EJTAGBEN       | —              | —              | POSCAGC        | ---            | POSCAGCDLY<1:0> |               |
| 23:16     | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P             | R/P           |
|           | POSCFGAIN<1:0> |                | POSCBOOST      | POSCGAIN<1:0>  |                | SOSCBOOST      | SOSCGAIN<1:0>   |               |
| 15:8      | R/P            | R/P            | R/P            | R/P            | r-y            | R/P            | R/P             | R/P           |
|           | SMCLR          | DBGPER<2:0>    |                |                | —              | FSLEEP         | FECCON<1:0>     |               |
| 7:0       | r-1            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P             | R/P           |
|           | —              | BOOTISA        | TRCEN          | ICESEL<1:0>    |                | JTAGEN         | DEBUG<1:0>      |               |

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit | P = Programmable bit               |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31 **Reserved:** The reset value of this bit is the same as DEVSIGN0<31>.

bit 30 **EJTAGBEN:** EJTAG Boot Enable bit

- 1 = Normal EJTAG functionality
- 0 = Reduced EJTAG functionality

bit 29-28 **Reserved:** Write as '1'

bit 27 **POSCAGC:** Primary Oscillator Automatic Gain Control bit

- 1 = Automatic gain control is enabled (default)
- 0 = Manual oscillator gain control

When the POSCAGC bit is enabled and POSC HS mode is selected, DEVCFG1<9:8> = '0b10' (i.e., POSCMOD), the Primary Oscillator will automatically do a linear search to find the lowest power/gain setting to guarantee oscillation with the users crystal.

**Note:** If the POSCMOD<1:0> bits (DEVCFG1/ADEVCFG1<9:8> = '0b00' (i.e., POSCMOD = EC mode), the POSCAGC bit must be set to '0'. POSCMOD = EC mode with POSCAGC = 1 is not permitted and will result in no oscillation.

bit 26 **Reserved:** Do not write a logic "0" to this bit.

bit 25-24 **POSCAGCDLY<1:0>:** Primary Crystal AGC Gain Search Step Settling Time Control bits

- 11 = Approximately (25 ms) (default)
- 10 = Approximately (6.25 ms)
- 01 = Approximately (400 ms)
- 00 = Approximately (100 ms)

**Note 1:** When the POSCAGC bit (DEVCFG0<27>) = 0 (i.e., manual oscillator gain control), these bits are not used. They are only used when AGC is enabled.

**2:** For POSC HS mode (DEVCFG1<9:8> = '0b10'), the default setting should meet the user crystal requirements. Internally, there are a maximum of 16 and a minimum of one AGC linear gain search steps the logic may utilize before locking. A lock will occur when the crystal is oscillating and the amplitude of the crystal signal is between a max and min fixed internal threshold. The POSCAGCDLY is the time for each of the possible AGC search steps settling time to allow the crystal to startup and amplitude stabilize before determining if a lock is true or to continue to search for the required gain. The POSCAGCDLY<1:0> bits represent a balance between start-up time and crystal power optimization. The lower the POSCAGCDLY delay time the faster the crystal start-up time but potentially at a higher crystal power level. The higher the POSCAGCDLY delay time the slower the crystal start-up time but with a better crystal power optimization level (i.e., less power).

**3:** Use of resonators with this product have not been confirmed -- use at your own discretion. When using a resonator, due to their long start-up times, it may be necessary to use a longer AGC gain step settling time.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 32-3: DEVCFG0/ADEVCFG0: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 0 (CONTINUED)

bit 23-22 **POSCFGAIN<1:0>**: Primary Crystal Oscillator Fine Gain Control bits

- 11 = Gain level 3 (highest, default)
- 10 = Gain is G2
- 01 = Gain is G1
- 00 = Gain is G0(lowest)

**Note 1:** G3 > G2 > G1 > G0.

**2:** When the POSCAGC bit (DEVCFG0<27>) = 1 (i.e., automatic gain control), or the POSCMOD<1:0> bits (DEVCFG1/ADEVCFG1<9:8>) ≠ '0b10 (i.e., HS crystal mode), the POSCGAIN<1:0> bits are not used.

**3:** These bits are used in conjunction with DEVCFG0/ADEVCFG0<20:19>. In almost all cases, the crystal fine gain default setting of '0b11 will work with the users course gain setting selection.

bit 21 **POSCBOOST**: Primary Oscillator Boost bit

- 1 = Uses internal XTAL feedback gain resistor (Default, in which case the user application should not use any external XTAL feedback resistor in the crystal circuit)
- 0 = Disconnects the internal XTAL feedback resistor

bit 20-19 **POSCGAIN<1:0>**: Primary Crystal Oscillator Coarse Gain Control bits

- 11 = Gain Level 3 (highest, default)
- 10 = Gain Level 2
- 01 = Gain Level 1
- 00 = Gain Level 0 (lowest)

**Note 1:** G3 > G2 > G1 > G0.

**2:** When the POSCAGC bit (DEVCFG0<27>) = 1 (i.e., automatic gain control), or the POSCMOD<1:0> bits (DEVCFG1/ADEVCFG1<9:8>) ≠ '0b10 (i.e., HS crystal mode), the POSCGAIN<1:0> bits are not used.

bit 18 **SOSCBOOST**: Secondary Oscillator Kick Start Programmability bit

- 1 = Start up and operate with high-power SOSC internal buffer only.
- 0 = Start up with internal SOSC high-power buffer, and then switch to low-power buffer when the SOSC is stable.

bit 17-16 **SOSCGAIN<1:0>**: Secondary Oscillator Gain Control bits

If SOSCGAIN<2> = 0:

- 11 = Gain is G3 (default)
- 10 = Gain is G2
- 01 = Gain is G1
- 00 = Gain is G0

**Note:** G3 > G2 > G1 > G0.

bit 15 **SMCLR**: Soft Master Clear Enable bit

- 1 = MCLR pin generates a normal system Reset
- 0 = MCLR pin generates a POR Reset

bit 14-12 **DBGPER<2:0>**: Debug Mode CPU Access Permission bits

- 1xx = Allow CPU access to Permission Group 2 permission regions
- x1x = Allow CPU access to Permission Group 1 permission regions
- xx1 = Allow CPU access to Permission Group 0 permission regions
- 0xx = Deny CPU access to Permission Group 2 permission regions
- x0x = Deny CPU access to Permission Group 1 permission regions
- xx0 = Deny CPU access to Permission Group 0 permission regions

**Note:** When the CPU is in Debug mode and the CPU1PG<1:0> bits (CFGPG<1:0>) are set to a denied permission group as defined by DBGPER<2:0>, the transaction request is assigned Group 3 permissions.

bit 11 **Reserved**: This bit is controlled by debugger/emulator development tools and should not be modified by the user.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 32-3: DEVCFG0/ADEVCFG0: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 0 (CONTINUED)

- bit 10 **FSLEEP:** Flash Sleep Mode bit  
1 = Flash is powered down when the device is in Sleep mode  
0 = Flash power down is controlled by the VREGS bit (PWRCON<0>)
- bit 9-8 **FECCCON<1:0>:** Dynamic Flash ECC Configuration bits  
11 = ECC and dynamic ECC are disabled (ECCCON<1:0> bits are writable)  
10 = ECC and dynamic ECC are disabled (ECCCON<1:0> bits are locked)  
01 = Dynamic Flash ECC is enabled (ECCCON<1:0> bits are locked)  
00 = Flash ECC is enabled (ECCCON<1:0> bits are locked; disables word Flash writes)  
**Note:** Upon a device POR, the value of these bits are copied by hardware into CFGCON<5:4> bits, (i.e. ECCCON<1:0>).
- bit 7 **Reserved:** Write as '1'
- bit 6 **BOOTISA:** Boot ISA Selection bit  
1 = Boot code and Exception code is MIPS32  
(ISAONEXC bit is set to '0' and the ISA<1:0> bits are set to '10' in the CP0 Config3 register)  
0 = Boot code and Exception code is microMIPS  
(ISAONEXC bit is set to '1' and the ISA<1:0> bits are set to '11' in the CP0 Config3 register)
- bit 5 **TRCEN:** Trace Enable bit  
1 = Trace features in the CPU are enabled  
0 = Trace features in the CPU are disabled
- bit 4-3 **ICESEL<1:0>:** In-Circuit Emulator/Debugger Communication Channel Select bits  
11 = PGEC1/PGED1 pair is used  
10 = PGEC2/PGED2 pair is used  
01 = PGEC3/PGED3 pair is used  
00 = Reserved
- bit 2 **JTAGEN:** JTAG Enable bit  
1 = JTAG is enabled  
0 = JTAG is disabled  
**Note 1:** On Reset, this Configuration bit is copied into JTAGEN (CFGCON<3>). If JTAGEN (DEVCFG0<2>) = 0, the JTAGEN bit cannot be set to '1' by the user application at run-time, as JTAG is always disabled. However, if JTAGEN (DEVCFG0<2>) = 1, the user application may enable/disable JTAG at run-time by simply writing JTAGEN (CFGCON<3>) as required.  
**2:** This bit sets the value of the JTAGEN bit in the CFGCON register.
- bit 1-0 **DEBUG<1:0>:** Background Debugger Enable bits (forced to '11' if code-protect is enabled)  
11 = 4-wire JTAG Enabled - PGECx/PGEDx Disabled - ICD module Disabled  
10 = 4-wire JTAG Enabled - PGECx/PGEDx Disabled - ICD module Enabled  
01 = PGECx/PGEDx Enabled - 4-wire JTAG I/F Disabled - ICD module Disabled  
00 = PGECx/PGEDx Enabled - 4-wire JTAG I/F Disabled - ICD module Enabled  
**Note:** When the FJTAGEN or JTAGEN bits are equal to '0', this prevents 4-wire JTAG debugging, but not PGECx/PGEDx debugging.

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 32-4: DEVCFG1/ADEVCFG1: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 1**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1  | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|
| 31:24     | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           |
|           | FDMTEN         | DMTCNT<4:0>    |                |                |                |                | FWDTWINSZ<1:0> |               |
| 23:16     | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           |
|           | FWDTEN         | WINDIS         | WDTSPGM        | WDTPS<4:0>     |                |                |                |               |
| 15:8      | R/P            | R/P            | r-1            | r-1            | r-1            | R/P            | R/P            | R/P           |
|           | FCKSM<1:0>     |                | —              | —              | —              | OSCIOFNC       | POSCMOD<1:0>   |               |
| 7:0       | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           |
|           | IESO           | FSOSCEN(1)     | DMTINV<2:0>    |                |                | FNOSC<2:0>     |                |               |

|                   |                  |  |
|-------------------|------------------|--|
| <b>Legend:</b>    | r = Reserved bit | P = Programmable bit                         |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

- bit 31     **FDMTEN:** Deadman Timer enable bit  
1 = Deadman Timer is enabled and *cannot* be disabled by software  
0 = Deadman Timer is disabled and *can* be enabled by software
- bit 30-26 **DMTCNT<4:0>:** Deadman Timer Count Select bits  
11111 = Reserved  
.  
.  
11000 = Reserved  
10111 = 2<sup>31</sup> (2147483648)  
10110 = 2<sup>30</sup> (1073741824)  
10101 = 2<sup>29</sup> (536870912)  
10100 = 2<sup>28</sup> (268435456)  
.  
.  
00001 = 2<sup>9</sup> (512)  
00000 = 2<sup>8</sup> (256)
- bit 25-24 **FWDTWINSZ<1:0>:** Watchdog Timer Window Size bits  
11 = Window size is 25%  
10 = Window size is 37.5%  
01 = Window size is 50%  
00 = Window size is 75%
- bit 23     **FWDTEN:** Watchdog Timer Enable bit  
1 = Watchdog Timer is enabled and cannot be disabled by software  
0 = Watchdog Timer is not enabled; it can be enabled in software
- bit 22     **WINDIS:** Watchdog Timer Window Enable bit  
1 = Watchdog Timer is in non-Window mode  
0 = Watchdog Timer is in Window mode
- bit 21     **WDTSPGM:** Watchdog Timer Stop During Flash Programming bit  
1 = Watchdog Timer stops during Flash programming  
0 = Watchdog Timer runs during Flash programming (for read/execute while programming Flash applications)

**Note 1:** If using external clock oscillator for SOSC instead of crystal, the FSOSCEN bit must be set to '0' with clock oscillator input connected to SOSCO; SOSC output pin not the SOSCI input pin. This will free up the SOSCI pin for use as an extra I/O pin.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 32-4: DEVCFG1/ADEVCFG1: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 1 (CONTINUED)

bit 20-16 **WDTPS<4:0>**: Watchdog Timer Postscale Select bits

10100 = 1:1048576  
10011 = 1:524288  
10010 = 1:262144  
10001 = 1:131072  
10000 = 1:65536  
01111 = 1:32768  
01110 = 1:16384  
01101 = 1:8192  
01100 = 1:4096  
01011 = 1:2048  
01010 = 1:1024  
01001 = 1:512  
01000 = 1:256  
00111 = 1:128  
00110 = 1:64  
00101 = 1:32  
00100 = 1:16  
00011 = 1:8  
00010 = 1:4  
00001 = 1:2  
00000 = 1:1

All other combinations not shown result in operation = 10100

bit 15-14 **FCKSM<1:0>**: Clock Switching and Monitoring Selection Configuration bits

11 = Software run-time clock switching is enabled and clock monitoring is enabled  
10 = Software run-time clock switching is disabled and clock monitoring is enabled  
01 = Software run-time clock switching is enabled and clock monitoring is disabled  
00 = Software run-time clock switching is disabled and clock monitoring is disabled

bit 13-11 **Reserved**: Write as '1'

bit 10 **OSCIOFNC**: CLKO Enable Configuration bit

1 = CLKO output is disabled  
0 = CLKO output signal is active on the OSC2 pin; Primary Oscillator must be disabled or configured for the External Clock mode (EC) for the CLKO to be active (POSCMOD<1:0> = 11 or 00)

bit 9-8 **POSCMOD<1:0>**: Primary Oscillator Configuration Mode bits

11 = Posc is disabled  
10 = HS Oscillator mode is selected  
01 = Reserved  
00 = EC mode is selected. This mode must not be selected if the POSCAGC bit (DEVCFG0/ADEVCFG0<27>) = 1.

bit 7 **IESO**: Internal External Switchover bit

1 = Internal to External HDW Clock Switchover mode is enabled (Two-Speed Start-up is enabled)  
0 = Internal to External HDW Clock Switchover mode is disabled (Two-Speed Start-up is disabled)

When IESO is set, CPU hardware will start up executing code on FRC and automatically switch to the clock source defined by FNOSC when that oscillator source is ready and stable, regardless of whether or not FCKSM clock switching is enabled.

bit 6 **FSOSCEN**: Secondary Oscillator Enable bit

1 = Enable Sosc  
0 = Disable Sosc

**Note 1:** If using external clock oscillator for SOSC instead of crystal, FSOSCEN bit must be set to '0' with clock oscillator input connected to SOSCO, SOSC output pin not the SOSCI input pin. This will free up the SOSCI pin for use as an extra I/O pin.



# PIC32MK GPG/MCJ with CAN FD Family

---

---

## REGISTER 32-4: DEVCFG1/ADEVCFG1: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 1 (CONTINUED)

bit 5-3 **DMTINV<2:0>**: Deadman Timer Count Window Interval bits

- 111 = Window/Interval value is 127/128 counter value
- 110 = Window/Interval value is 63/64 counter value
- 101 = Window/Interval value is 31/32 counter value
- 100 = Window/Interval value is 15/16 counter value
- 011 = Window/Interval value is 7/8 counter value
- 010 = Window/Interval value is 3/4 counter value
- 001 = Window/Interval value is 1/2 counter value
- 000 = Window/Interval value is zero

bit 2-0 **FNOSC<2:0>**: Oscillator Selection bits

- 111 = Reserved
- 110 = Backup Fast RC (BFRC) Oscillator
- 101 = Low-Power RC Oscillator (LPRC)
- 100 = Secondary Oscillator (Sosc)
- 011 = Reserved
- 010 = Primary Oscillator (Posc) (HS, EC)
- 001 = System PLL (SPLL Module) (input clock and divider set by SPLLCON)
- 000 = Fast RC Oscillator (FRC) divided by the FRCDIV<2:0> bits (OSCCON<26:24>)  
(supports FRC / n, where n = 1, 2, 4, 8, 16, 32, 64, 256)

**Note 1:** If using external clock oscillator for SOSC instead of crystal, FSOSCEN bit must be set to '0' with clock oscillator input connected to SOSCO, SOSC output pin not the SOSCI input pin. This will free up the SOSCI pin for use as an extra I/O pin.

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 32-5: DEVCFG2/ADEVCFG2: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 2

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-1            | r-1            | R/P            | R/P-0          | R/P-0          | R/P-0          | R/P-0         | R/P-0         |
|           | —              | —              | BORSEL         | —              | —              | —              | —             | —             |
| 23:16     | R/P-0          | R/P-0          | R/P-0          | R/P-0          | r-1            | R/P            | R/P           | R/P           |
|           | —              | —              | —              | —              | —              | FPLL0DIV<2:0>  |               |               |
| 15:8      | r-1            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           | R/P           |
|           | —              | FPLLMULT<6:0>  |                |                |                |                |               |               |
| 7:0       | R/P            | R/P            | R/P            | R/P            | r-1            | R/P            | R/P           | R/P           |
|           | FPLLICK        | FPLL RNG<2:0>  |                |                | —              | FPLLIDIV<2:0>  |               |               |

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit | P = Programmable bit               |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31-30 **Reserved:** Write as '1'

bit 29 **BORSEL:** Brown-out Reset (BOR) Select Trip Voltage bit  
 1 = BOR trip voltage 2.168V (non-op amp device operation)  
 0 = BOR trip voltage 2.932V (op amp device operation)

**Note:** The user application must select the greatest BORSEL voltage to enable the highest trip voltage possible that is still less than VDD application operating voltage.

bit 28-20 **Reserved:** Write as '0'

bit 19 **RESERVED:** Write as "1"

bit 18-16 **FPLL0DIV<2:0>:** Default System PLL Output Divisor bits

- 111 = PLL output divided by 32
- 110 = PLL output divided by 32
- 101 = PLL output divided by 32
- 100 = PLL output divided by 16
- 011 = PLL output divided by 8
- 010 = PLL output divided by 4
- 001 = PLL output divided by 2
- 000 = PLL output divided by 2

bit 15 **Reserved:** Write as '1'

bit 14-8 **FPLLMULT<6:0>:** System PLL Feedback Divider bits

- 1111111 = Multiply by 128
- 1111110 = Multiply by 127
- 1111101 = Multiply by 126
- 1111100 = Multiply by 125
- .
- .
- .
- 0000000 = Multiply by 1

bit 7 **FPLLICK:** System PLL Input Clock Select bit  
 1 = FRC is selected as input to the System PLL  
 0 = POSC is selected as input to the System PLL

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 32-5: DEVCFG2/ADEVCFG2: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 2 (CONTINUED)

bit 6-4 **FPLL RNG<2:0>**: System PLL Divided Input Clock Frequency Range bits

111 = Reserved  
110 = Reserved  
101 = 34-64 MHz  
100 = 21-42 MHz  
011 = 13-26 MHz  
010 = 8-16 MHz  
001 = 5-10 MHz  
000 = Bypass

**Note:** Use the highest filter range that covers the input frequency to the VCO multiplier block that corresponds to the PLLIDIV output freq to minimize PLL system jitter (see [FIGURE 7-1: "PIC32MK GPG/MCJ with CAN FD Family Oscillator Diagram"](#)). For example, Crystal = 20 MHz, PLLIDIV<2:0> = '0b1; therefore, the filter input frequency is equal to 10 MHz and SPLLRANGE FPLL RNG<2:0> = '0b010.

bit 3 **Reserved:** Write as '1'

bit 2-0 **FPLLIDIV<2:0>**: PLL Input Divider bits

111 = Divide by 8  
110 = Divide by 7  
101 = Divide by 6  
100 = Divide by 5  
011 = Divide by 4  
010 = Divide by 3  
001 = Divide by 2  
000 = Divide by 1

# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 32-6: DEVCFG3/ADEVCFG3: DEVICE/ALTERNATE DEVICE CONFIGURATION WORD 3

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | r-1            | r-1            | R/P            | R/P            | R/P            | r-1            | r-1           | r-1           |
|           | —              | —              | IOL1WAY        | PMDL1WAY       | PGL1WAY        | —              | —             | —             |
| 23:16     | r-1            | r-1            | r-1            | R/P            | r-1            | r-1            | r-1           | r-1           |
|           | —              | —              | —              | PWMLOCK        | —              | —              | —             | —             |
| 15:8      | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           | R/P           |
|           | USERID<15:8>   |                |                |                |                |                |               |               |
| 7:0       | R/P            | R/P            | R/P            | R/P            | R/P            | R/P            | R/P           | R/P           |
|           | USERID<7:0>    |                |                |                |                |                |               |               |

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| <b>Legend:</b>    | r = Reserved bit | P = Programmable bit               |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 31-30 **RESERVED:** Write as "1"

bit 29 **IOL1WAY:** Peripheral Pin Select Configuration bit

- 1 = Allow only one reconfiguration
- 0 = Allow multiple reconfigurations

bit 28 **PMDL1WAY:** Peripheral Module Disable Configuration bit

- 1 = Allow only one reconfiguration
- 0 = Allow multiple reconfigurations

bit 27 **PGL1WAY:** Permission Group Lock One Way Configuration bit

- 1 = Allow only one reconfiguration
- 0 = Allow multiple reconfigurations

bit 26-22 **Reserved:** Write as '1'

bit 21 **Reserved:** Write as '1'

bit 20 **PWMLOCK:** PWM Write Access Select bit

- 1 = Write accesses to the PWM IOCONx register are not locked or protected
- 0 = Write accesses to the PWM IOCONx register must use the PWMKEY unlock procedure

bit 19-16 **Reserved:** Write as '1'

bit 15-0 **USERID<15:0>:** This is a 16-bit value that is user-defined and is readable via ICSP™ and JTAG

# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 32-7: CFGCON: CONFIGURATION CONTROL REGISTER**

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5        | Bit 28/20/12/4         | Bit 27/19/11/3        | Bit 26/18/10/2        | Bit 25/17/9/1         | Bit 24/16/8/0         |
|-----------|----------------|----------------|-----------------------|------------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 31:24     | U-0            | U-0            | U-0                   | U-0                    | U-0                   | R/W-0                 | r-0                   | U-0                   |
|           | —              | —              | —                     | —                      | —                     | ADCPRI <sup>(1)</sup> | —                     | —                     |
| 23:16     | R/W-0          | R/W-0          | R/W-0                 | R/W-0                  | R/W-0                 | R/W-0                 | R/W-0                 | R/W-0                 |
|           | —              | —              | —                     | —                      | —                     | —                     | ICACLK <sup>(1)</sup> | OCACLK <sup>(1)</sup> |
| 15:8      | U-0            | U-0            | R/W-0                 | R/W-0                  | R/W-0                 | r-0                   | r-0                   | U-0                   |
|           | —              | —              | IOLOCK <sup>(1)</sup> | PMDLOCK <sup>(1)</sup> | PGLOCK <sup>(1)</sup> | —                     | —                     | —                     |
| 7:0       | U-0            | U-0            | R/W-y                 | R/W-y                  | R/W-y                 | R/W-0                 | U-0                   | R/W-1                 |
|           | —              | —              | ECCCON<1:0>           |                        | JTAGEN                | TROEN                 | —                     | TDOEN                 |

|                   |                  |  |
|-------------------|------------------|--|
| <b>Legend:</b>    | r = Reserved bit | y = Value set from Configuration bits on POR |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-27 **Unimplemented:** Read as '0'

bit 26 **ADCPRI:** ADC Arbitration Priority to SRAM bit<sup>(1)</sup>

1 = ADC gets High Priority access to SRAM

0 = ADC uses Least Recently Serviced Arbitration (same as other initiators)

bit 25 **Reserved:** Write as '0'

bit 24-18 **Unimplemented:** Read as '0'

bit 17 **ICACLK:** Input Capture Alternate Clock Selection bit<sup>(1)</sup>

1 = Input Capture modules use an alternative Timer pair as their timebase clock

0 = All Input Capture modules use Timer2/3 as their timebase clock

bit 16 **OCACLK:** Output Compare Alternate Clock Selection bit<sup>(1)</sup>

1 = Output Compare modules use an alternative Timer pair as their timebase clock

0 = All Output Compare modules use Timer2/3 as their timebase clock

bit 15-14 **Unimplemented:** Read as '0'

bit 13 **IOLOCK:** Peripheral Pin Select Lock bit<sup>(1)</sup>

1 = Peripheral Pin Select is locked. Writes to PPS registers are not allowed

0 = Peripheral Pin Select is not locked. Writes to PPS registers are allowed

bit 12 **PMDLOCK:** Peripheral Module Disable bit<sup>(1)</sup>

1 = Peripheral module is locked. Writes to PMD registers are not allowed

0 = Peripheral module is not locked. Writes to PMD registers are allowed

bit 11 **PGLOCK:** Permission Group Lock bit<sup>(1)</sup>

1 = Permission Group registers are locked. Writes to PG registers are not allowed

0 = Permission Group registers are not locked. Writes to PG registers are allowed

bit 10-9 **Reserved:** Write as '0'

bit 8 **Unimplemented:** Read as '0'

bit 7-6 **Unimplemented:** Read/write as '0'

bit 5-4 **ECCCON<1:0>:** Flash ECC Configuration bits

11 = ECC and dynamic ECC are disabled (ECCCON<1:0> bits are writable)

10 = ECC and dynamic ECC are disabled (ECCCON<1:0> bits are locked)

01 = Dynamic Flash ECC is enabled (ECCCON<1:0> bits are locked)

00 = Flash ECC is enabled (ECCCON<1:0> bits are locked; disables word Flash writes)

**Note:** These bits are loaded from DEVCFG0<9:8>, (i.e. FECCCON<1:0>), configuration word fuse bits on POR power on reset

**Note 1:** To change this bit, the unlock sequence must be performed. Refer to **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the *"PIC32 Family Reference Manual"* for details.

# PIC32MK GPG/MCJ with CAN FD Family

---

## REGISTER 32-7: CFGCON: CONFIGURATION CONTROL REGISTER (CONTINUED)

bit 3 **JTAGEN:** JTAG Port Enable bit

- 1 = Enable the JTAG port
- 0 = Disable the JTAG port

**Note:** The reset value of this bit is the value of the JTAGEN Configuration Word setting in the DEVCFG0 register. If JTAGEN (DEVCFG0<2>) = 0, this bit cannot be set to '1' by the user application at run-time. If JTAGEN (DEVCFG0<2>) = 1, the user application may enable/disable JTAG at run-time by writing this bit to the desired value.

bit 2 **TROEN:** Trace Output Enable bit

- 1 = Enable trace outputs and start trace clock (trace probe must be present)
- 0 = Disable trace outputs and stop trace clock

**Note:** When the user Configuration Word, TRCEN in the DEVCFG0 register is equal to '0', the value of this bit is ignored, but has the effect of being '0'.

bit 1 **Unimplemented:** Read as '0'

bit 0 **TDOEN:** TDO Enable for 2-Wire JTAG

- 1 = 2-wire JTAG protocol uses TDO
- 0 = 2-wire JTAG protocol does not use TDO

**Note:** Implementing the JTAG protocol over the 2-wire interface requires four 2-wire clocks for each TCK if TDO is required. However, if the values shifted out TDO are predetermined, TDO can be disabled.

**Note 1:** To change this bit, the unlock sequence must be performed. Refer to **Section 42. "Oscillators with Enhanced PLL"** (DS60001250) in the *"PIC32 Family Reference Manual"* for details.



# PIC32MK GPG/MCJ with CAN FD Family

## REGISTER 32-9: DEVID: DEVICE AND REVISION ID REGISTER

| Bit Range | Bit 31/23/15/7              | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3              | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|-----------------------------|----------------|----------------|----------------|-----------------------------|----------------|---------------|---------------|
| 31:24     | R                           | R              | R              | R              | R                           | R              | R             | R             |
|           | VER<3:0> <sup>(1)</sup>     |                |                |                | DEVID<27:24> <sup>(1)</sup> |                |               |               |
| 23:16     | R                           | R              | R              | R              | R                           | R              | R             | R             |
|           | DEVID<23:16> <sup>(1)</sup> |                |                |                |                             |                |               |               |
| 15:8      | R                           | R              | R              | R              | R                           | R              | R             | R             |
|           | DEVID<15:8> <sup>(1)</sup>  |                |                |                |                             |                |               |               |
| 7:0       | R                           | R              | R              | R              | R                           | R              | R             | R             |
|           | DEVID<7:0> <sup>(1)</sup>   |                |                |                |                             |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-28 **VER<3:0>**: Revision Identifier bits<sup>(1)</sup>

bit 27-0 **DEVID<27:0>**: Device ID<sup>(1)</sup>

**Note 1:** See the "PIC32 Flash Programming Specification" (DS60001145) for a list of Revision and Device ID values.

## REGISTER 32-10: DEVADCx: DEVICE ADC CALIBRATION REGISTER 'x' ('x' = 0-5, 7)

| Bit Range | Bit 31/23/15/7 | Bit 30/22/14/6 | Bit 29/21/13/5 | Bit 28/20/12/4 | Bit 27/19/11/3 | Bit 26/18/10/2 | Bit 25/17/9/1 | Bit 24/16/8/0 |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 31:24     | R              | R              | R              | R              | R              | R              | R             | R             |
|           | ADCAL<31:24>   |                |                |                |                |                |               |               |
| 23:16     | R              | R              | R              | R              | R              | R              | R             | R             |
|           | ADCAL<23:16>   |                |                |                |                |                |               |               |
| 15:8      | R              | R              | R              | R              | R              | R              | R             | R             |
|           | ADCAL<15:8>    |                |                |                |                |                |               |               |
| 7:0       | R              | R              | R              | R              | R              | R              | R             | R             |
|           | ADCAL<7:0>     |                |                |                |                |                |               |               |

### Legend:

R = Readable bit                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -n = Value at POR                      '1' = Bit is set                      '0' = Bit is cleared                      x = Bit is unknown

bit 31-0 **ADCAL<31:0>**: Calibration Data for the ADC Module bits

Before enabling the ADC, the user application must initialize the ADC calibration values by copying them from the factory programmed DEVADCx Flash locations starting at 0xBFC45000 into the ADCxCFG registers starting at 0xBF887D00, respectively. Refer to [23.0 "12-bit High-Speed Successive Approximation Register \(SAR\) Analog-to-Digital Converter \(ADC\)"](#) for more information.



# PIC32MK GPG/MCJ with CAN FD Family

**REGISTER 32-11: DEVSNx: DEVICE SERIAL NUMBER REGISTER 'x' ('x' = 0-3)**

| Bit Range | Bit<br>31/23/15/7 | Bit<br>30/22/14/6 | Bit<br>29/21/13/5 | Bit<br>28/20/12/4 | Bit<br>27/19/11/3 | Bit<br>26/18/10/2 | Bit<br>25/17/9/1 | Bit<br>24/16/8/0 |
|-----------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 31:24     | R                 | R                 | R                 | R                 | R                 | R                 | R                | R                |
|           | SN<31:24>         |                   |                   |                   |                   |                   |                  |                  |
| 23:16     | R                 | R                 | R                 | R                 | R                 | R                 | R                | R                |
|           | SN<23:16>         |                   |                   |                   |                   |                   |                  |                  |
| 15:8      | R                 | R                 | R                 | R                 | R                 | R                 | R                | R                |
|           | SN<15:8>          |                   |                   |                   |                   |                   |                  |                  |
| 7:0       | R                 | R                 | R                 | R                 | R                 | R                 | R                | R                |
|           | SN<7:0>           |                   |                   |                   |                   |                   |                  |                  |

**Legend:**

|                   |                  |  |
|-------------------|------------------|--|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0'           |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared      x = Bit is unknown |

bit 31-0      **SN<31:0>**: Device Unique Serial Number bits

These registers contain a value, programmed during factory production test, that is unique to each unit and are user read only. These values are persistent and not erased even when a new application code is programmed into the device. These values can be used if desired as an encryption key in combination with the Microchip encryption library.

# PIC32MK GPG/MCJ with CAN FD Family

## 32.3 On-Chip Voltage Regulator

The core and digital logic for all PIC32MK GPG/MCJ with CAN FD Family of devices are designed to operate at a nominal 1.2V. To simplify system designs, devices in the PIC32MK GPG/MCJ with CAN FD Family incorporate an on-chip regulator providing the required core logic voltage from VDD.

### 32.3.1 ON-CHIP REGULATOR AND POR

It takes a fixed delay for the on-chip regulator to generate an output. During this time, designated as TPU, code execution is disabled. TPU is applied every time the device resumes operation after any power-down, including Sleep mode.

### 32.3.2 ON-CHIP REGULATOR AND BOR

PIC32MK GPG/MCJ with CAN FD Family of devices also have a simple brown-out capability. If the voltage supplied to the regulator is inadequate to maintain a regulated level, the regulator Reset circuitry will generate a Brown-out Reset. This event is captured by the BOR flag bit (RCON<1>). The brown-out voltage levels are specific in [36.1 “DC Characteristics”](#).

## 32.4 On-chip Temperature Sensor

PIC32MK GPG/MCJ with CAN FD Family of devices include a temperature sensor that provides accurate measurement of a device’s junction temperature (see [36.2 “AC Characteristics and Timing Parameters”](#) for more information).

The temperature sensor is connected to the ADC module and can be measured using the shared S&H circuit (see [23.0 “12-bit High-Speed Successive Approximation Register \(SAR\) Analog-to-Digital Converter \(ADC\)”](#) for more information).

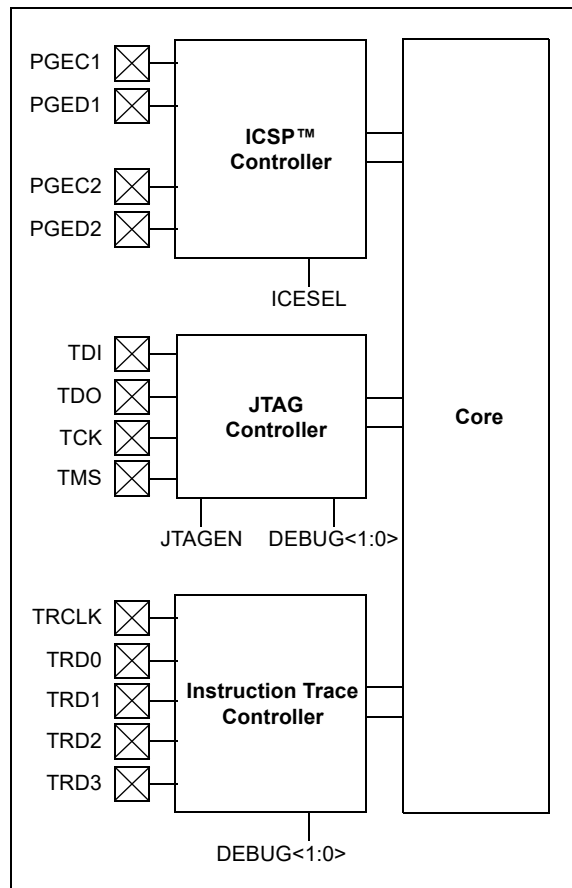
## 32.5 Programming and Diagnostics

PIC32MK GPG/MCJ with CAN FD Family of devices provide a complete range of programming and diagnostic features that can increase the flexibility of any application using them. These features allow system designers to include:

- Simplified field programmability using two-wire In-Circuit Serial Programming™ (ICSP™) interfaces
- Debugging using ICSP
- Programming and debugging capabilities using the EJTAG extension of JTAG
- JTAG boundary scan testing for device and board diagnostics

PIC32MK devices incorporate two programming and diagnostic modules, and a trace controller, that provide a range of functions to the application developer.

FIGURE 32-1: BLOCK DIAGRAM OF PROGRAMMING, DEBUGGING AND TRACE PORTS



# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

---

## 33.0 INSTRUCTION SET

The PIC32MK GPG/MCJ with CAN FD Family instruction set complies with the MIPS32<sup>®</sup> Release 5 instruction set architecture. The PIC32MK GPG/MCJ with CAN FD Family device family *does not* support the following features:

- Core extend instructions
- Coprocessor 1 instructions
- Coprocessor 2 instructions

**Note:** Refer to “MIPS32<sup>®</sup> Architecture for Programmers Volume II: The MIPS32<sup>®</sup> Instruction Set” at [www.imgtec.com](http://www.imgtec.com) for more information.

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

---

---

# PIC32MK GPG/MCJ with CAN FD Family

## 34.0 MIGRATION GUIDE

TABLE 34-1: PIC32MKXXGPD/GPEXX TO PIC32MKXXGPGXX MIGRATION REFERENCE

| PIC32MKxxGPD/GPExx                                  | FEATURE                          | PIC32MKxxGPGxx  | 64-pin PIC32MKxxGPD/E<br>to<br>64-pin PIC32MKxxGPG<br>Migration Impact |                  |
|---|----------------------------------|---|--|------------------|
|   |                                  |   | PCB HDW  | SW               |
| 100/64  | #PIN                             | 64/44   | X  | X                |
| 77 (100-pin), 48 (64-pin)                           | # I/O                            | 53 (64-pin), 37 (48-pin)                                    | X  | X                |
| MIPS arch @ 120 MHz                                 | CPU                              | MIPS arch @ 120 MHz   | ---  | ---              |
| 4   | Configuration Word Registers     | 5   | ---  | X                |
| 1024 / 512 Mb<br>No ECC<br>Live Update<br>Dual Boot | FLASH                            | 524,288,262,144 Mb<br>ECC<br>No Live Update<br>No Dual Boot | ---  | X <sup>(2)</sup> |
| 256, 128 Kb   | SRAM                             | 64-Kb Only  | ---  | X <sup>(1)</sup> |
| POSC<br>SOSC<br>UPLL (USB PLL)                      | Oscillator                       | POSC w/AGC & Fine gain<br>SOSC<br>No UPLL (No USB)          | X <sup>(2)</sup>   | X <sup>(2)</sup> |
| 4-Kb  | EE Data Flash Module             | (None)  | ---  | X                |
| (None)  | CLC<br>(Configurable Logic Cell) | 4   | ---  | ---              |
| 8/13  | DMA                              | 8/2   | ---  | X                |
| (2) Full-Speed (100 pin)<br>(1) Full-Speed (64 pin) | USB                              | (None)  | X  | X                |
| 1   | TIMER1                           | 1 (Identical)   | ---  | ---              |
| 8   | 32 bit TIMERS 2-9 type B         | 8 (Identical)   | ---  | ---              |
| 1   | Watch Dog Timer                  | 1 (identical)   | ---  | ---              |
| 1   | Dead Man Timer                   | 1   | ---  | ---              |
| 16  | Input Capture                    | 8   | X <sup>(1)</sup>   | X <sup>(1)</sup> |
| 16  | Output Compare                   | 8   | X <sup>(1)</sup>   | X <sup>(1)</sup> |
| 6   | SPI                              | 2   | X <sup>(1)</sup>   | X <sup>(1)</sup> |
| (None)  | I <sup>2</sup> C                 | 2   | ---  | ---              |
| 6   | UART                             | 2   | X <sup>(1)</sup>   | X <sup>(1)</sup> |
| 1   | PMP                              | (None)  | X  | X                |
| 1   | RTCC                             | 1 (Identical)   | ---  | ---              |
| CAN (Lite)  | CAN                              | CAN FD<br>(New HDW/SW)                                      | ---  | X                |
| 7   | ADC Modules                      | 7 (Identical)   | ---  | ---              |
| 42 (100-pin), 26 (64 pin-)                          | ADC Channels                     | 30 (64-pin), 18 (48-pin)                                    | X <sup>(1)</sup>   | X <sup>(1)</sup> |
| 4   | OPAMP                            | 4 (New features<br>and configurations)                      | ---  | X                |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 34-1: PIC32MKXXGPD/GPEXX TO PIC32MKXXGPGXX MIGRATION REFERENCE (CONTINUED)**

| PIC32MKxxGPD/GPExx       | FEATURE                         | PIC32MKxxGPGxx                      | 64-pin PIC32MKxxGPD/E to 64-pin PIC32MKxxGPG Migration Impact |                  |
|--------------------------|---------------------------------|-------------------------------------|---|------------------|
|                          |                                 |                                     | PCB HDW   | SW               |
| 5                        | COMPARATORS                     | 5 (New features and configurations) | ---   | X                |
| 3                        | DAC                             | 2                                   | X <sup>(1)</sup>  | X <sup>(1)</sup> |
| (None)                   | Low Voltage Detect              | 1                                   | ---   | X                |
| 1                        | CTMU                            | 1 (Identical)                       | ---   | ---              |
| Yes (20)                 | CVD Module (Touch Enhancement)  | Yes (24)                            | X <sup>(1)</sup>  | X <sup>(1)</sup> |
| Yes                      | PMD (Peripheral Module Disable) | Yes (Fewer Peripherals)             | ---   | X <sup>(1)</sup> |
| Yes                      | JTAG                            | Yes                                 | ---   | ---              |
| Yes (100-pin and 64-pin) | TRACE                           | Yes (64-pin), No (48-pin)           | ---   | ---              |
| 3                        | PGCx/PGDx (Debug)               | 3                                   | ---   | ---              |
| Yes                      | VBAT                            | No                                  | X   | X                |
| Yes                      | Deep Sleep                      | No                                  | ---   | X                |

**Note 1:** This is only affected if the user application is using any additional modules, peripheral functions, peripheral pin function, or features on the PIC32MKxxGPD/GPExx that the PIC32MKxxGPGxx does not possess (see [Table 34-2](#) for more information).

**2:** POSC Legacy code on PIC32MKxxGPD/GPExx will default in silicon on to PIC32MKxxGPGxx POSC w/ AGC and ignore POSC gain setting. If using an external shunt gain resistor across POSC XTAL on PIC32MKxxGPD/GPExx users must ensure that the DEVCFG0<POSCBOOST>=0 on PIC32MKxxGPGxx or remove the resistor for operation. Either internal or external gain boost but not both.



**TABLE 34-2: 64 PIN PIC32MKXXGPD/E TO PIC32MKXXGPG FUNCTION MIGRATION MISMATCHES**

| Pin # | PIC32MKxxGPD/GPExx 64 pin                                   | PIC32MKxxGPGxx 64 Pin                                  | Pin Function Match | Mismatch Function | Peripheral |
|-------|---|--|--------------------|-------------------|------------|
| 1     | TCK/RPA7/PMD5/RA7   | TCK/RPA7/RA7   | NO                 | PMD5              | PMP        |
| 2     | RPB14/VBUSON1/PMD6/RB14                                     | RPB14/RB14   | NO                 | VBUSON1/PMD6      | USB / PMP  |
| 3     | RPB15/PMD7/RB15   | RPB15/RB15   | NO                 | PMD7              | PMP        |
| 4     | AN19/CVD19/RPG6/PMA5/RG6                                    | AN19/CVD19/RPG6/RG6                                    | NO                 | PMA5              | PMP        |
| 5     | AN18/CVD18/RPG7/PMA4/RG7(6)                                 | AN18/CVD18/RPG7/RG7                                    | NO                 | PMA4              | PMP        |
| 6     | AN17/CVD17/RPG8/PMA3/RG8(7)                                 | AN17/CVD17/RPG8/RG8                                    | NO                 | PMA3              | PMP        |
| 7     | MCLR#   | MCLR#  | MATCH              | ---               | ---        |
| 8     | AN16/CVD16/RPG9/PMA2/RG9                                    | AN16/CVD16/RPG9/RG9                                    | NO                 | PMA2              | PMP        |
| 9     | VSS   | VSS  | MATCH              | ---               | ---        |
| 10    | VDD   | VDD  | MATCH              | ---               | ---        |
| 11    | AN10/CVD10/RPA12/RA12                                       | AN10/CVD10/RPA12/RA12                                  | MATCH              | ---               | ---        |
| 12    | AN9/CVD9/RPA11/RA11   | AN9/CVD9/RPA11/RA11                                    | MATCH              | ---               | ---        |
| 13    | OA2OUT/AN0/C2IN4-/C4IN3-/RPA0/RA0                           | OA2OUT/AN0/C2IN4-/C4IN3-/RPA0/RA0                      | MATCH              | ---               | ---        |
| 14    | OA2IN+/AN1/C2IN1+/RPA1/RA1                                  | OA2IN+/AN1/C2IN1+/RPA1/RA1                             | MATCH              | ---               | ---        |
| 15    | PGD3/VREF-/OA2IN-/AN2/C2IN1-/RPB0/<br>CTED2/RB0             | PGD3/VREF-/OA2IN-/AN2/C2IN1-/RPB0/CTED2/<br>RB0        | MATCH              | ---               | ---        |
| 16    | PGC3/OA1OUT/VREF+/AN3/C1IN4-/C4IN2-/<br>RPB1/CTED1/PMA6/RB1 | PGC3/OA1OUT/VREF+/AN3/C1IN4-/C4IN2-/<br>RPB1/CTED1/RB1 | NO                 | PMA6              | PMP        |
| 17    | PGC1/OA1IN+/AN4/C1IN1+/C1IN3-/C2IN3-/<br>RPB2/RB2           | PGC1/OA1IN+/AN4/C1IN1+/C1IN3-/C2IN3-/RPB2/<br>RB2      | MATCH              | ---               | ---        |
| 18    | PGD1/OA1IN-/AN5/CTCMP/C1IN1-/RTCC/<br>RPB3/RB3              | PGD1/OA1IN-/AN5/CTCMP/C1IN1-/RTCC/RPB3/<br>RB3         | MATCH              | ---               | ---        |
| 19    | AVDD  | AVDD   | MATCH              | ---               | ---        |
| 20    | AVSS  | AVSS   | MATCH              | ---               | ---        |
| 21    | OA3OUT/AN6/CVD6/C3IN4-/C4IN1+/C4IN4-/<br>RPC0/RC0           | OA3OUT/AN6/CVD6/C3IN4-/C4IN1+/C4IN4-/<br>RPC0/RC0      | MATCH              | ---               | ---        |

**TABLE 34-2: 64 PIN PIC32MKXXGPD/E TO PIC32MKXXGPG FUNCTION MIGRATION MISMATCHES (CONTINUED)**

| Pin # | PIC32MKxxGPD/GPExx 64 pin                          | PIC32MKxxGPGxx 64 Pin                    | Pin Function Match | Mismatch Function                                | Peripheral |
|-------|--|--|--------------------|--|------------|
| 22    | OA3IN-/AN7/CVD7/C3IN1-/C4IN1-/RPC1/PMA7/RC1        | OA3IN-/AN7/CVD7/C3IN1-/C4IN1-/RPC1/RC1   | NO                 | PMA7   | PMP        |
| 23    | OA3IN+/AN8/CVD8/C3IN1+/C3IN3-/RPC2/PMA13/RC2       | OA3IN+/AN8/CVD8/C3IN1+/C3IN3-/RPC2/RC2   | NO                 | PMA13  | PMP        |
| 24    | AN11/CVD11/C1IN2-/PMA12/RC11                       | AN11/CVD11/C1IN2-/RC11                   | NO                 | PMA12  | PMP        |
| 25    | VSS  | VSS                                      | MATCH              | ---  | ---        |
| 26    | VDD  | VDD                                      | MATCH              | ---  | ---        |
| 27    | AN12/CVD12/C2IN2-/C5IN2-/PMA11/RE12                | AN12/CVD12/C2IN2-/C5IN2-/RE12            | NO                 | PMA11  | PMP        |
| 28    | AN13/CVD13/C3IN2-/PMA10/RE13                       | AN13/CVD13/C3IN2-/RE13                   | NO                 | PMA10  | PMP        |
| 29    | AN14/CVD14/RPE14/PMA1/RE14                         | AN14/CVD14/RPE14/RE14                    | NO                 | PMA1   | PMP        |
| 30    | AN15/CVD15/RPE15/PMA0/RE15                         | AN15/CVD15/RPE15/RE15                    | NO                 | PMA0   | PMP        |
| 31    | TDI/DAC3/AN26/CVD26/RPA8/PMA9/RA8                  | TDI/DAC2/AN26/CVD26/RPA8/SDA2/RA8        | NO                 | PMA9/DAC3  | PMP/DAC    |
| 32    | RPB4/PMA8/RB4                                      | RPB4/SCL2/RB4                            | NO                 | PMA8/SCL2  | PMP/I2C    |
| 33    | OA5IN+/DAC1/AN24/CVD24/C5IN1+/C5IN3-/RPA4/T1CK/RA4 | OA5IN+/AN24/CVD24/C5IN1+/C5IN3-/RPA4/RA4 | NO                 | T1CLK  | Timer1     |
| 34    | VBUS   | AN40/CVD40/RPE0/RE0                      | NO                 | ALL Functions:<br>USB / ANx / CVD /<br>RPxx / IO | ---        |
| 35    | VUSB3V3  | AN41/CVD41/RPE1/RE1                      | NO                 | ---  | ---        |
| 36    | D-   | AN46/CVD46/RPA14/RA14                    | NO                 | ---  | ---        |
| 37    | D+   | AN47/CVD47/RPA15/RA15                    | NO                 | ---  | ---        |
| 38    | VDD  | VDD                                      | MATCH              | ---  | ---        |
| 39    | OSCI/CLKI/AN49/CVD49/RPC12/RC12                    | OSCI/CLKI/AN49/CVD49/RPC12/RC12          | MATCH              | ---  | ---        |
| 40    | OSCO/CLKO/RPC15/RC15                               | OSCO/CLKO/RPC15/RC15                     | MATCH              | ---  | ---        |
| 41    | VSS  | VSS                                      | MATCH              | ---  | ---        |
| 42    | VBAT   | RD8                                      | NO                 | VBAT /RD8  | VBAT & I/O |
| 43    | PGD2/RPB5/USBID1/RB5                               | PGD2/RPB5/SDA1/RB5                       | NO                 | SDA1 / USBID                                     | I2C / USB  |

**TABLE 34-2: 64 PIN PIC32MKXXGPD/E TO PIC32MKXXGPG FUNCTION MIGRATION MISMATCHES (CONTINUED)**

| Pin # | PIC32MKxxGPD/GPExx 64 pin                       | PIC32MKxxGPGxx 64 Pin                           | Pin Function Match | Mismatch Function    | Peripheral  |
|-------|---|---|--------------------|----------------------|-------------|
| 44    | PGC2/RPB6/SCK2/PMA15/RB6                        | PGC2/RPB6/SCL1/RB6                              | NO                 | SCLK2/PMA15/<br>SCL1 | SPI/PMP/I2C |
| 45    | DAC2/AN48/CVD48/RPC10/PMA14/RC10                | DAC1/AN48/CVD48/RPC10/RC10                      | NO                 | PMA14/DAC2           | PMP/DAC     |
| 46    | OA5OUT/AN25/CVD25/C5IN4-/RPB7/SCK1/<br>INT0/RB7 | OA5OUT/AN25/CVD25/C5IN4-/RPB7/SCK1/INT0/<br>RB7 | MATCH              | ---                  | ---         |
| 47    | SOSCI/RPC13/RC13                                | SOSCI/RPC13/RC13                                | NO                 | ---                  | ---         |
| 48    | SOSCO/RPB8/RB8                                  | SOSCO/RPB8/T1CK/RB8                             | NO                 | T1CLK                | Timer1      |
| 49    | TMS/OA5IN-/AN27/CVD27/C5IN1-/RPB9/RB9           | TMS/OA5IN-/AN27/CVD27/LVDIN/C5IN1-/RPB9/<br>RB9 | NO                 | LVDIN                | HLVD        |
| 50    | TRCLK/RPC6/RC6                                  | TRCLK/RPC6/RC6                                  | MATCH              | ---                  | ---         |
| 51    | TRD0/RPC7/RC7                                   | TRD0/RPC7/RC7                                   | MATCH              | ---                  | ---         |
| 52    | TRD1/RPC8/PMWR/RC8                              | TRD1/RPC8/RC8                                   | NO                 | PMWR                 | PMP         |
| 53    | TRD2/RPD5/PMRD/RD5                              | TRD2/RPD5/RD5                                   | NO                 | PMRD                 | PMP         |
| 54    | TRD3/RPD6/RD6                                   | TRD3/RPD6/RD6                                   | MATCH              | ---                  | ---         |
| 55    | RPC9/RC9  | RPC9/RC9  | MATCH              | ---                  | ---         |
| 56    | VSS   | VSS   | MATCH              | ---                  | ---         |
| 57    | VDD   | VDD   | MATCH              | ---                  | ---         |
| 58    | RPF0/RF0  | RPF0/RF0  | MATCH              | ---                  | ---         |
| 59    | RPF1/RF1  | RPF1/RF1  | MATCH              | ---                  | ---         |
| 60    | RPB10/PMD0/RB10                                 | RPB10/RB10                                      | NO                 | PMD0                 | PMP         |
| 61    | RPB11/PMD1/RB11                                 | RPB11/RB11                                      | NO                 | PMD1                 | PMP         |
| 62    | RPB12/PMD2/RB12                                 | RPB12/RB12                                      | NO                 | PMD2                 | PMP         |
| 63    | RPB13/CTPLS/PMD3/RB13                           | RPB13/CTPLS/RB13                                | NO                 | PMD3                 | PMP         |
| 64    | TDO/PMD4/RA10                                   | TDO/RA10  | NO                 | PMD4                 | PMP         |

# PIC32MK GPG/MCJ with CAN FD Family

---

## 35.0 DEVELOPMENT SUPPORT

The PIC<sup>®</sup> microcontrollers (MCU) and dsPIC<sup>®</sup> digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB<sup>®</sup> X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM<sup>™</sup> Assembler
  - MPLINK<sup>™</sup> Object Linker/  
MPLIB<sup>™</sup> Object Librarian
  - MPLAB Assembler/Linker/Librarian for  
Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE<sup>™</sup> In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICKit<sup>™</sup> 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards,  
Evaluation Kits and Starter Kits
- Third-party development tools

## 35.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows<sup>®</sup>, Linux and Mac OS<sup>®</sup> X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

# PIC32MK GPG/MCJ with CAN FD Family

---

## 35.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

## 35.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

## 35.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

## 35.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

# PIC32MK GPG/MCJ with CAN FD Family

---

## 35.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

## 35.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

## 35.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

## 35.9 PICkit 3 In-Circuit Debugger/Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

## 35.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

# PIC32MK GPG/MCJ with CAN FD Family

---

## 35.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) for the complete list of demonstration, development and evaluation kits.

## 35.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

# PIC32MK GPG/MCJ with CAN FD Family

## 36.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of the PIC32MK GPG/MCJ with CAN FD Family electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the PIC32MK GPG/MCJ with CAN FD Family devices are listed below. Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these or any other conditions, above the parameters indicated in the operation listings of this specification, is not implied.

### Absolute Maximum Ratings

(See Note 1)

|   |                      |
|---|----------------------|
| Ambient temperature under bias .....  | <-40°C to +125°C     |
| Storage temperature .....   | -65°C to +150°C      |
| Voltage on VDD with respect to VSS .....  | -0.3V to +4.0V       |
| Voltage on any pin that is not 5V tolerant, with respect to VSS (Note 3) .....    | -0.3V to (VDD +0.3V) |
| Voltage on any 5V tolerant pin with respect to VSS when VDD ≥ 2.3V (Note 3) ..... | -0.3V to +5.5V       |
| Voltage on any 5V tolerant pin with respect to VSS when VDD < 2.3V (Note 3) ..... | -0.3V to +3.6V       |
| Voltage on VBUS with respect to VSS .....   | -0.3V to +5.5V       |
| Maximum current out of VSS pin(s) .....   | 200 mA               |
| Maximum current into VDD pin(s) (Note 2) .....                                    | 200 mA               |
| Maximum current sunk/sourced by any 4x I/O pin (Note 4) .....                     | 15 mA                |
| Maximum current sunk/sourced by any 8x I/O pin (Note 4) .....                     | 25 mA                |
| Maximum current sunk by all ports .....   | 150 mA               |
| Maximum current sourced by all ports (Note 2) .....                               | 150 mA               |
| Maximum junction temperature .....  | -40°C to +140°C      |

### ESD qualification:

|  |             |
|--|-------------|
| Human Body Model (HBM) per JESD22-A114 .....   | 2000 V      |
| Machine Model (MM) per JESD22-A115 .....   | 200 V       |
| Charged Device Model (CDM) AEC Q100-011 (ANSI/ESD STM 5.3.1)(Middle / Corner pins) ..... | 500 / 750 V |

- Note 1:** Stresses above those listed under “**Absolute Maximum Ratings**” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions, above those indicated in the operation listings of this specification, is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
- 2:** Maximum allowable current is a function of device maximum power dissipation (see [Table 36-2](#)).
- 3:** See the pin name tables ([Table 2](#) and [Table 4](#)) for the 5V tolerant pins.
- 4:** Characterized, but not tested. Refer to parameters [DO10](#), [DO20](#), and [DO20a](#) for the 4x and 8x I/O pin lists.



# PIC32MK GPG/MCJ with CAN FD Family

## 36.1 DC Characteristics

**TABLE 36-1: OPERATING MIPS VERSUS VOLTAGE**

| Characteristic | VDD Range (in Volts) (Note 1) | Temp. Range (in °C) | Max. Frequency                             | Comment    |
|----------------|-------------------------------|---------------------|--|------------|
|                |                               |                     | PIC32MK GPG/MCJ with CAN FD Family Devices |            |
| DC5a           | 2.3V-3.6V <sup>(1,2)</sup>    | -40°C to +85°C      | 120 MHz                                    | Industrial |
| DC5b           | 3.0 V-3.6V <sup>(1,3)</sup>   |                     |  |            |
| DC5c           | 2.3V-3.6V <sup>(1,2)</sup>    | -40°C to +125°C     | 80 MHz                                     | Extended   |
| DC5d           | 3.0V-3.6V <sup>(1,3)</sup>    |                     |  |            |

**Note 1:** Overall functional device operation at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  is guaranteed, but not characterized. All device analog modules, such as ADC, etc., will function, but with degraded performance below  $V_{DDMIN}$ . Refer to parameter BO10 in [Table 36-5](#) for BOR values.

- 2:** If  $DEVCFG2<BORSEL> = 1$ , (Set to "1" if op amps are not being utilized in application,  $V_{DD} (min) = 2.3v$ )  
**3:** If  $DEVCFG2<BORSEL> = 0$  (Required if op amps are being utilized in application,  $V_{DD} (min) = 3.0v$ )

**TABLE 36-2: THERMAL OPERATING CONDITIONS**

| Rating  | Symbol | Min.                      | Typ. | Max. | Unit |
|---|--------|---------------------------|------|------|------|
| <b>Industrial Temperature Devices</b>   |        |                           |      |      |      |
| Junction Temperature Range <sup>(1)</sup>   | TJ     | (1)                       | —    | (1)  | °C   |
| Operating Ambient Temperature Range   | TA     | -40                       | —    | +85  | °C   |
| <b>Extended Temperature Devices</b>   |        |                           |      |      |      |
| Junction Temperature Range <sup>(1)</sup>   | TJ     | (1)                       | —    | (1)  | °C   |
| Operating Ambient Temperature Range   | TA     | -40                       | —    | +125 | °C   |
| Power Dissipation:<br>Internal Chip Power Dissipation:<br>$P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$<br>I/O Pin Power Dissipation:<br>$P_{I/O} = \sum ((V_{DD} - V_{OH}) \times I_{OH}) + (V_{OL} \times I_{OL})$ | PD     | PINT + PI/O               |      |      | W    |
| Maximum Allowed Power Dissipation   | PDMAX  | $(T_J - T_A)/\theta_{JA}$ |      |      | W    |

**Note 1:** See [Absolute Maximum Ratings](#).

**TABLE 36-3: THERMAL PACKAGING CHARACTERISTICS**

| Characteristics                                      | Symbol        | Typ. | Max. | Unit | Notes |
|--|---------------|------|------|------|-------|
| Package Thermal Resistance, 64-pin QFN (9x9x0.9 mm)  | $\theta_{JA}$ | 28   | —    | °C/W | 1     |
| Package Thermal Resistance, 64-pin TQFP (10x10x1 mm) | $\theta_{JA}$ | 38.7 | —    | °C/W | 1     |
| Package Thermal Resistance, 48-pin TQFP (7x7x1 mm)   | $\theta_{JA}$ | 55.6 | —    | °C/W | 1     |
| Package Thermal Resistance, 48-pin VQFN (6x6x0.9 mm) | $\theta_{JA}$ | 21.8 | —    | °C/W | 1     |

**Note 1:** Junction to ambient thermal resistance, Theta-JA ( $\theta_{JA}$ ) numbers are achieved by package simulations.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS**

| DC CHARACTERISTICS       |          |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |                           |       |  |
|--------------------------|----------|---|---|------|---------------------------|-------|--|
| Param. No.               | Symbol   | Characteristics   | Min.  | Typ. | Max.                      | Units | Conditions   |
| <b>Operating Voltage</b> |          |   |   |      |                           |       |  |
| DC10a                    | VDD(1,4) | <b>Supply Voltage</b>   | 2.3   | —    | 3.6                       | V     | DEVCFG2<BORSEL> = 1  |
| DC10b                    | VDD(1,5) |   | 3.0   | —    | 3.6                       | V     | DEVCFG2<BORSEL> = 0  |
| DC12                     | VDR      | <b>RAM Data Retention Voltage (Note 2)</b>  | 1.75  | —    | —                         | V     | —  |
| DC16                     | VPOR     | <b>VDD Start Voltage</b><br>to Ensure Internal<br>Power-on Reset Signal ( <b>Note 3</b> ) | —   | —    | V <sub>SS</sub> +<br>0.3V | V     | Non-compliance can<br>cause indeterminate<br>behavior and/or start-<br>up issues |
| DC17                     | SVDD     | <b>VDD Rise Rate</b><br>to Ensure Internal<br>Power-on Reset Signal                       | 0.000011  | —    | 0.33                      | V/μs  | 300 ms to 10μs   |

- Note 1:** Overall functional device operation at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  is guaranteed, but not characterized. All device Analog modules, such as ADC etc., will function, but with degraded performance below  $V_{DDMIN}$ . Refer to parameter BO10 in [Table 36-5](#) for BOR values.
- 2:** This is the limit to which VDD can be lowered without losing RAM data.
- 3:** This is the limit to which VDD must be lowered to ensure Power-on Reset.
- 4:** If DEVCFG2<BORSEL> = 1, (Set to "1" if op amps are not being utilized in application,  $V_{DD(min)} = 2.3v$ )
- 5:** If DEVCFG2<BORSEL> = 0 (Required if op amps are being utilized in application,  $V_{DD(min)} = 3.0v$ )

**TABLE 36-5: ELECTRICAL CHARACTERISTICS: BOR**

| DC CHARACTERISTICS |        |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |       |       |                     |
|--------------------|--------|--|---|------|-------|-------|---------------------|
| Param. No.         | Symbol | Characteristics  | Min. <sup>(1)</sup>   | Typ. | Max.  | Units | Conditions          |
| BO10a              | VBOR   | BOR Event on VDD transition<br>high-to-low ( <b>Note 2</b> ) | 2.638   | —    | 2.942 | V     | DEVCFG2<BORSEL> = 0 |
|                    |        |  | 1.928   | —    | 2.172 | V     | DEVCFG2<BORSEL> = 1 |

- Note 1:** Parameters are for design guidance only and are not tested in manufacturing.
- 2:** Overall functional device operation at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  is tested, but not characterized. All device Analog modules, such as ADC, etc., will function, but with degraded performance below  $V_{DDMIN}$ .

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-6: DC CHARACTERISTICS: OPERATING CURRENT (IDD RUN CURRENT WITH PERIPHERAL CLOCKS ENABLED)<sup>(1,2)</sup>**

| DC CHARACTERISTICS   |                          |         | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                    |
|--|--------------------------|---------|---|--------------------|
| Parameter No.  | Typical <sup>(3,4)</sup> | Maximum | Units   | Conditions         |
| <b>Operating Current (IDD Run Current With Peripheral Clocks Enabled) (Note 1,2)</b>           |                          |         |   |                    |
| DC20   | 3.4                      | 28      | mA  | 4 MHz (Note 2,4)   |
| DC21   | 5.4                      | 31      | mA  | 10 MHz (Note 2,4)  |
| DC22   | 18.7                     | 43      | mA  | 60 MHz (Note 2,4)  |
| DC23   | 24.4                     | 49      | mA  | 80 MHz (Note 2)    |
| DC25   | 36                       | 60      | mA  | 120 MHz (Note 2)   |
| <b>Operating Current (IDD CPU Only Run Current With Peripheral Clocks Disabled) (Note 1,2)</b> |                          |         |   |                    |
| DC20A  | 2.9                      | 28      | mA  | 4 MHz (Note 4,5)   |
| DC21A  | 4                        | 30      | mA  | 10 MHz (Note 4,5)  |
| DC22A  | 13                       | 39      | mA  | 60 MHz (Note 4,5)  |
| DC23A  | 17                       | 43      | mA  | 80 MHz (Note 4,5)  |
| DC25A  | 25                       | 51      | mA  | 120 MHz (Note 4,5) |

**Note 1:** A device's IDD supply current is mainly a function of the operating voltage and frequency. Other factors, such as Peripheral Bus Clock (PBCLK) frequency, number of peripheral modules enabled, internal code execution pattern, I/O pin loading and switching rate, oscillator type, as well as temperature, can have an impact on the current consumption.

**2:** The test conditions for IDD measurements are as follows:

- Oscillator mode is EC (for 8 MHz and below) and EC+PLL (for above 8 MHz) with OSC1 driven by external square wave from rail-to-rail, (OSC1 input clock input over/undershoot < 100 mV required)
- OSC2/CLKO is configured as an I/O input pin
- PBCLKx divisor = 1:2 ('x' ≠ 1,7), PBCLK6 = 1:4, PBCLK1 = 1:1
- CPU, Program Flash, and SRAM data memory are operational, Program Flash memory Wait states are equal to seven (default)
- Prefetch module is enabled
- No peripheral modules are operating, (ON bit = 0), and the associated PMD bit is '0' (clocks enabled)
- WDT, DMT, Clock Switching, Fail-Safe Clock Monitor, and Secondary Oscillator are disabled
- All I/O pins are configured as inputs and pulled to Vss
- MCLR = VDD
- CPU executing `while(1)` statement from Flash
- RTCC and JTAG are disabled

**3:** Data in the "Typical" column is at 3.3V, +25°C at specified operating frequency unless otherwise stated. Parameters are for design guidance only and are not tested.

**4:** This parameter is characterized, but not tested in manufacturing.

**5:** Note 2 applies with the following exceptions:

- Prefetch disabled
- Prefetch cache disabled
- PMDx = 1 (all bits set)
- PB2, 3, 4, 5, 6 = OFF
- PB1 = 1:128

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-7: DC CHARACTERISTICS: IDLE CURRENT (IDLE)**

| DC CHARACTERISTICS   |                           |         | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                          |
|--|---------------------------|---------|---|--------------------------|
| Parameter No.  | Typical <sup>(2, 3)</sup> | Maximum | Units   | Conditions               |
| <b>Idle Current (IDLE): Core Off, Clock on Base Current (Note 1)</b> |                           |         |   |                          |
| DC30a  | 2.7                       | 27      | mA  | 4 MHz ( <b>Note 3</b> )  |
| DC31a  | 3.7                       | 29      | mA  | 10 MHz                   |
| DC32a  | 11                        | 35      | mA  | 60 MHz ( <b>Note 3</b> ) |
| DC33a  | 17                        | 43      | mA  | 120 MHz                  |

**Note 1:** The test conditions for IDLE current measurements are as follows:

- Oscillator mode is EC (for 8 MHz and below) and EC+PLL (for above 8 MHz) with OSC1 driven by external square wave from rail-to-rail, (OSC1 input clock input over/undershoot < 100 mV required)
  - OSC2/CLKO is configured as an I/O input pin
  - PBCLKx divisor = 1:2 ('x' ≠ 1,6,7), PBCLK6 = 1:4, PBCLK1 = 1:1
  - CPU is in Idle mode (CPU core Halted)
  - Prefetch module is disabled
  - No peripheral modules are operating, (ON bit = 0), and the associated PMD bit is '0' (i.e., clocks enabled)
  - WDT, DMT, Clock Switching, Fail-Safe Clock Monitor, and Secondary Oscillator are disabled
  - All I/O pins are configured as inputs and pulled to Vss
  - $\overline{\text{MCLR}} = \text{VDD}$
  - RTCC and JTAG are disabled
- 2:** Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
- 3:** This parameter is characterized, but not tested in manufacturing.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-8: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)**

| DC CHARACTERISTICS                       |                        | Standard Operating Conditions: 2.3V to 3.6V(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |       |            |  |
|--|------------------------|---|-------|------------|--|
| Param. No.                               | Typical <sup>(2)</sup> | Maximum   | Units | Conditions |  |
| <b>Power-Down Current (IPD) (Note 1)</b> |                        |   |       |            |  |
| DC40k                                    | 0.4                    | 3   | mA    | -40°C      | Base Power-Down Sleep                                      |
| DC40l                                    | 0.7                    | 3   | mA    | +25°C      |  |
| DC40m                                    | 2.5                    | 9.5   | mA    | +85°C      |  |
| DC40o                                    | 7                      | 26  | mA    | +125°C     |  |
| <b>Module Differential Current</b>       |                        |   |       |            |  |
| DC41e                                    | 5                      | —   | μA    | 3.6V       | Watchdog Timer Current: $\Delta I_{WDT}$ (Note 3)          |
| DC42e                                    | 25                     | —   | μA    | 3.6V       | RTCC + Timer1 w/32 kHz Crystal: $\Delta I_{RTCC}$ (Note 3) |
| DC43d                                    | 4                      | 6   | mA    | 3.6V       | ADC: $\Delta I_{ADC}$ (Notes 3, 4)                         |

**Note 1:** The test conditions for IPD current measurements are as follows:

Sleep:

- Oscillator mode is EC (for 8 MHz and below) and EC+PLL (for above 8 MHz) with OSC1 driven by external square wave from rail-to-rail, (OSC1 input clock input over/undershoot < 100 mV required)
  - OSC2/CLKO is configured as an I/O input pin
  - PBCLKx divisor = 1:2 ('x' ≠ 1,6,7), PBCLK6 = 1:4, PBCLK1 = 1:1
  - CPU is in Sleep mode
  - Prefetch module is disabled
  - No peripheral modules are operating, (ON bit = 0), and the associated PMD bit is '0' (i.e., clocks enabled)
  - WDT, DMT, Clock Switching, Fail-Safe Clock Monitor, and Secondary Oscillator are disabled
  - All I/O pins are configured as inputs and pulled to Vss
  - $\overline{\text{MCLR}} = V_{DD}$
  - RTCC and JTAG are disabled
  - Voltage regulator is in Stand-by mode (VREGS = 0)
- 2:** Data in the "Typical" column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
- 3:** The  $\Delta$  current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.
- 4:** Voltage regulator is operational (VREGS = 1)

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS**

| DC CHARACTERISTICS                   |                       |  | Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |                            |        |   |
|--------------------------------------|-----------------------|--|--|---------------------|----------------------------|--------|---|
| Param. No.                           | Symbol                | Characteristics  | Min.   | Typ. <sup>(1)</sup> | Max.                       | Units  | Conditions  |
| DI10                                 | V <sub>IL</sub>       | <b>Input Low Voltage</b>                                   |  |                     |                            |        |   |
|                                      |                       | SDAx, SCLx   | V <sub>SS</sub>  | —                   | 0.3 V <sub>DD</sub>        | V      | SMBus disabled  |
|                                      |                       | SDAx, SCLx<br>I/O Pins                                     | V <sub>SS</sub><br>V <sub>SS</sub>   | —<br>—              | 0.8<br>0.2 V <sub>DD</sub> | V<br>V | SMBus enabled   |
| DI20                                 | V <sub>IH</sub>       | <b>Input High Voltage</b>                                  |  |                     |                            |        |   |
|                                      |                       | I/O Pins not 5V-tolerant <sup>(5)</sup>                    | 0.65*V <sub>DD</sub>   | —                   | V <sub>DD</sub>            | V      | SMBus disabled<br>SMBus enabled, 2.3V = < VPIN = < 5.5                          |
|                                      |                       | SDAx, SCLx   | 0.65*V <sub>DD</sub>   | —                   | <b>(7)</b>                 | V      |   |
|                                      |                       | SDAx, SCLx   | 2.1  | —                   | <b>(7)</b>                 | V      |   |
| I/O Pins 5V-tolerant <sup>(5)</sup>  | 0.65* V <sub>DD</sub> | —  | 5.5  | V                   |                            |        |   |
| DI30                                 | ICNPU                 | <b>Change Notification Pull-up Current</b>                 | -450   | —                   | -50                        | μA     | V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>SS</sub><br><b>(Note 3,6)</b> |
| DI31                                 | ICNPD                 | <b>Change Notification Pull-down Current<sup>(4)</sup></b> | 50   | —                   | 450                        | μA     | V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>DD</sub>                      |
| DI50<br>DI51<br>DI53<br>DI55<br>DI56 | I <sub>IL</sub>       | <b>Input Leakage Current (Note 3)</b>                      |  |                     |                            |        |   |
|                                      |                       | I/O Ports  | -1   | —                   | 1                          | μA     | V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> ,<br>Pin at high-impedance |
|                                      |                       | ADC and VREF Input Pins                                    | -1   | —                   | 1                          | μA     | V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> ,<br>Pin at high-impedance |
|                                      |                       | Op amp Input Shared Pins                                   | -35  | —                   | 35                         | μA     |   |
|                                      |                       | MCLR <sup>(2)</sup>  | -1   | —                   | 1                          | μA     | V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub>                            |
|                                      |                       | OSC1   | -1   | —                   | 1                          | μA     | V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> ,<br>HS mode               |

- Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
- 2:** The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3:** Negative current is defined as current sourced by the pin.
- 4:** This parameter is characterized, but not tested in manufacturing.
- 5:** Refer to the pin name tables (Table 2 and Table 4) for the 5V-tolerant pins.
- 6:** The V<sub>IH</sub> specifications are only in relation to externally applied inputs, and not with respect to the user-selectable internal MCU VDD pull-ups. External open drain input signals utilizing the internal pull-ups of the PIC32 device are guaranteed to be recognized only as a logic “high” internally to the PIC32 device, provided that the external load does not exceed the minimum value of ICNPU. For external 5V tolerant “input” logic inputs that require a pull-up source, to guarantee the minimum V<sub>IH</sub> of those components, it is recommended to use an external pull-up resistor rather than the internal MCU VDD pull-ups of the PIC32 device.
- 7:** V<sub>IH</sub> (maximum) for Non-5V tolerant I<sup>2</sup>C pin is V<sub>DD</sub>. V<sub>IH</sub> (maximum) for 5V tolerant I<sup>2</sup>C pins is 5.5V.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-10: DC CHARACTERISTICS: I/O PIN INPUT INJECTION CURRENT SPECIFICATIONS**

| DC CHARACTERISTICS |                   |  | Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |                       |       |  |
|--------------------|-------------------|--|--|---------------------|-----------------------|-------|--|
| Param. No.         | Symbol            | Characteristics  | Min.   | Typ. <sup>(1)</sup> | Max.                  | Units | Conditions   |
| DI60a              | I <sub>ICL</sub>  | <b>Input Low Injection Current</b>                                     | 0  | —                   | -5 <sup>(2,5)</sup>   | mA    | This parameter applies to all pins, with the exception of RB10. Maximum I <sub>ICL</sub> current for this exception is 0 mA.   |
| DI60b              | I <sub>ICH</sub>  | <b>Input High Injection Current</b>                                    | 0  | —                   | +5 <sup>(3,4,5)</sup> | mA    | This parameter applies to all pins, with the exception of all 5V tolerant pins, SOSCI, SOSCO, OSC1, OSC2, D-, D+, RTCC, and RB10. Maximum I <sub>ICH</sub> current for these exceptions is 0 mA. |
| DI60c              | ∑I <sub>ICT</sub> | <b>Total Input Injection Current (sum of all I/O and control pins)</b> | -20 <sup>(6)</sup>   | —                   | +20 <sup>(6)</sup>    | mA    | Absolute instantaneous sum of all ± input injection currents from all I/O pins<br>(   I <sub>ICL</sub>   +   I <sub>ICH</sub>   ) ≤ ∑I <sub>ICT</sub>  |

- Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
- 2:** V<sub>IL</sub> source < (V<sub>SS</sub> - 0.3). Characterized but not tested.
- 3:** V<sub>IH</sub> source > (V<sub>DD</sub> + 0.3) for non-5V tolerant pins only.
- 4:** Digital 5V tolerant pins do not have an internal high side diode to V<sub>DD</sub>, and therefore, cannot tolerate any “positive” input injection current.
- 5:** Injection currents > | 0 | can affect the ADC results by approximately 4 to 6 counts (i.e., V<sub>IH</sub> Source > (V<sub>DD</sub> + 0.3) or V<sub>IL</sub> source < (V<sub>SS</sub> - 0.3)).
- 6:** Any number and/or combination of I/O pins not excluded under I<sub>ICL</sub> or I<sub>ICH</sub> conditions are permitted provided the “absolute instantaneous” sum of the input injection currents from all pins do not exceed the specified limit. If **Note 2**, I<sub>ICL</sub> = ((V<sub>SS</sub> - 0.3) - V<sub>IL</sub> source) / R<sub>s</sub>. If **Note 3**, I<sub>ICH</sub> = ((I<sub>CH</sub> source - (V<sub>DD</sub> + 0.3)) / R<sub>S</sub>). R<sub>S</sub> = Resistance between input source voltage and device pin. If (V<sub>SS</sub> - 0.3) ≤ V<sub>SOURCE</sub> ≤ (V<sub>DD</sub> + 0.3), injection current = 0.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-11: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS**

| DC CHARACTERISTICS |      |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |      |      |       |   |
|--------------------|------|---|---|------|------|-------|---|
| Param.             | Sym. | Characteristic  | Min.  | Typ. | Max. | Units | Conditions  |
| DO10               | VOL  | <b>Output Low Voltage</b><br>I/O Pins<br>4x Sink Driver Pins -<br>RA0, RA4, RA11, RA12, RA14, RA15<br>RB0-RB3, RB8, RB9<br>RC0-RC2, RC10, RC12, RC13<br>RD8<br>RE0, RE1<br>RG8, RG9     | —   | —    | 0.4  | V     | $I_{OL} \leq 10 \text{ mA}$ , $V_{DD} = 3.3\text{V}$  |
|                    |      | <b>Output Low Voltage</b><br>I/O Pins:<br>8x Sink Driver Pins -<br>RA1, RA7, RA8, RA10<br>RB4-RB7, RB10-RB15<br>RC6-RC9, RC11, RC15<br>RD5, RD6<br>RE12-RE15<br>RF0, RF1<br>RG6, RG7    | —   | —    | 0.4  | V     | $I_{OL} \leq 15 \text{ mA}$ , $V_{DD} = 3.3\text{V}$  |
| DO20               | VOH  | <b>Output High Voltage</b><br>I/O Pins:<br>4x Source Driver Pins -<br>RA0, RA4, RA11, RA12, RA14, RA15<br>RB0-RB3, RB8, RB9<br>RC0-RC2, RC10, RC12, RC13<br>RD8<br>RE0, RE1<br>RG8, RG9 | 2.4   | —    | —    | V     | $I_{OH} \geq -10 \text{ mA}$ , $V_{DD} = 3.3\text{V}$ |
|                    |      | <b>Output High Voltage</b><br>I/O Pins:<br>8x Source Driver Pins -<br>RA1, RA7, RA8, RA10<br>RB4-RB7, RB10-RB15<br>RC6-RC9, RC11, RC15<br>RD5, RD6<br>RE12-RE15<br>RF0, RF1<br>RG6, RG7 | 2.4   | —    | —    | V     | $I_{OH} \geq -15 \text{ mA}$ , $V_{DD} = 3.3\text{V}$ |



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-11: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS (CONTINUED)**

| DC CHARACTERISTICS |      |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |      |       |                          |
|--------------------|------|--|---|------|------|-------|--------------------------|
| Param.             | Sym. | Characteristic   | Min.  | Typ. | Max. | Units | Conditions               |
| DO20a              | VOH1 | <b>Output High Voltage</b><br>I/O Pins:<br>4x Source Driver Pins -<br>RA0, RA4, RA11, RA12, RA14, RA15<br>RB0-RB3, RB8, RB9<br>RC0-RC2, RC10, RC12, RC13<br>RD8<br>RE0, RE1<br>RG8, RG9                            | 1.5   | —    | —    | V     | IOH ≥ -14 mA, VDD = 3.3V |
|                    |      |  | 2.0   | —    | —    | V     | IOH ≥ -12 mA, VDD = 3.3V |
|                    |      |  | 3.0   | —    | —    | V     | IOH ≥ -7 mA, VDD = 3.3V  |
|                    | VOH1 | <b>Output High Voltage</b><br>I/O Pins:<br>8x Source Driver Pins -<br>8x Source Driver Pins -<br>RA1, RA7, RA8, RA10<br>RB4-RB7, RB10-RB15<br>RC6-RC9, RC11, RC15<br>RD5, RD6<br>RE12-RE15<br>RF0, RF1<br>RG6, RG7 | 1.5   | —    | —    | V     | IOH ≥ -22 mA, VDD = 3.3V |
|                    |      |  | 2.0   | —    | —    | V     | IOH ≥ -18 mA, VDD = 3.3V |
|                    |      |  | 3.0   | —    | —    | V     | IOH ≥ -10 mA, VDD = 3.3V |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-12: DC CHARACTERISTICS: PROGRAM MEMORY<sup>(3)</sup>**

| DC CHARACTERISTICS |       |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |        |            |   |
|--------------------|-------|--|---|---------------------|--------|------------|---|
| Param. No.         | Sym.  | Characteristics  | Min.  | Typ. <sup>(1)</sup> | Max.   | Units      | Conditions                                      |
| D130               | EP    | Cell Endurance   | 20,000  | —                   | —      | E/W        | —   |
| D131               | VPR   | VDD for Read   | VDDMIN  | —                   | VDDMAX | V          | —   |
| D132               | VPEW  | VDD for Erase or Write   | VDDMIN  | —                   | VDDMAX | V          | —   |
| D134               | TRETD | Characteristic Retention   | 20  | —                   | —      | Year       | —   |
| D135               | IDDP  | Supply Current during Programming  | —   | —                   | 30     | mA         | —   |
| D136               | TRW   | Row Write Cycle Time ( <b>Notes 2</b> )  | 16000   | —                   | 20800  | FRC Cycles | —   |
| D137               | TQWW  | Quad Word Write Cycle Time   | 650   | —                   | 777    | FRC Cycles | —   |
| D138               | TWW   | Word Write Cycle Time  | 265   | —                   | 297    | FRC Cycles | —   |
| D139               | TCE   | Chip Erase Cycle Time ( <b>Note 4</b> )  | 384000  | —                   | 480000 | FRC Cycles | —   |
| D140               | TPFE  | Combined Upper Plus Lower Flash Panels Erase Cycle Time (both Boot Flash excluded)           | 256000  | —                   | 320000 | FRC Cycles | —   |
| D141               | TPBE  | Single Panel Flash Erase Cycle Time (either Upper or Lower Panel, excluding both Boot Flash) | 128000  | —                   | 160000 | FRC Cycles | —   |
| D142               | TPGE  | Page Erase Cycle Time  | 128000  | —                   | 160000 | FRC Cycles | Erase Retry Disabled - VREAD1 (NVMCON2<12>) = 0 |
|                    |       |  | 32000   | —                   | 160000 | FRC Cycles | Erase Retry enabled - VREAD1 (NVMCON2<12>) = 1  |
| D143               | TFLPU | NVM Power-up Delay   | —   | —                   | 10     | µs         | —   |

**Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated.

**2:** The minimum SYSCLK for row programming is 4 MHz.

**3:** Refer to the “PIC32 Flash Programming Specification” (DS60001145) for operating conditions during programming and erase cycles.

**4:** This parameter depends on FRC accuracy (see [Table 36-17](#)) and FRC tuning values (see the OSCTUN register: [Register 7-2](#)).

**TABLE 36-13: DC CHARACTERISTICS: PROGRAM FLASH MEMORY WAIT STATES FOR ACTIVE HIGH POWER MODE**

| DC CHARACTERISTICS   |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |  |
|--|---|---|--|
| DEVCFG0<9:8>   | Required Flash Wait States<br>PFMWS<3:0> bits | SYSCLK (MHz)  |  |
| FECCON<1:0> bits<br>(DEVCFG0<9:8>) = 0x1x<br>with ECC disabled | 1 - Wait State                                | 0 < SYSCLK ≤ 116 MHz  |  |
|  | 2 - Wait State                                | 116 MHz < SYSCLK ≤ 120 MHz  |  |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-13: DC CHARACTERISTICS: PROGRAM FLASH MEMORY WAIT STATES FOR ACTIVE HIGH POWER MODE (CONTINUED)**

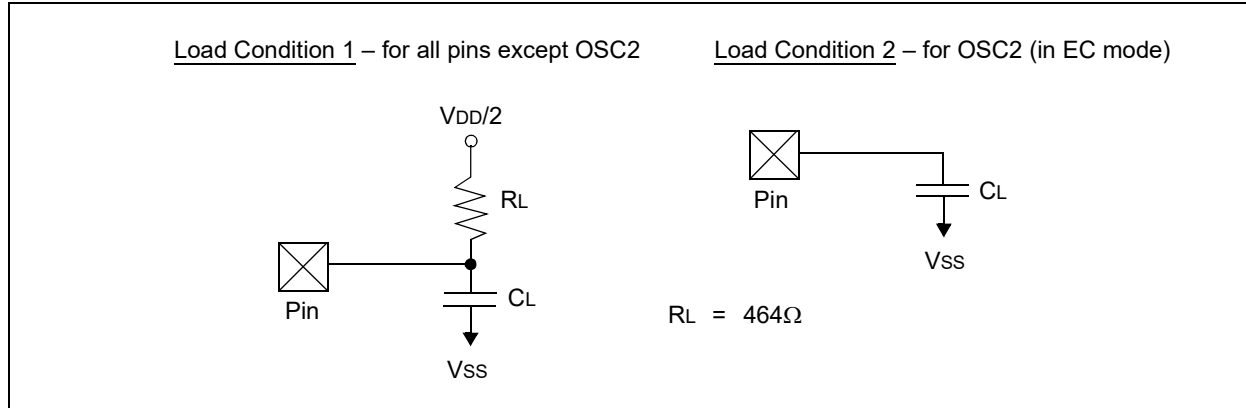
| DC CHARACTERISTICS   |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |
|--|---|---|
| DEVCFG0<9:8>   | Required Flash Wait States<br>PFMWS<3:0> bits | SYSCLK (MHz)  |
| FECCCON<1:0> bits<br>(DEVCFG0<9:8>) = 0x0x<br>with ECC enabled | 1 - Wait State                                | $0 < \text{SYSCLK} \leq 96 \text{ MHz}$   |
|  | 2 - Wait State                                | $96 \text{ MHz} < \text{SYSCLK} \leq 120 \text{ MHz}$   |

# PIC32MK GPG/MCJ with CAN FD Family

## 36.2 AC Characteristics and Timing Parameters

The information contained in this section defines PIC32MK GPG/MCJ with CAN FD Family device AC characteristics and timing parameters.

**FIGURE 36-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**



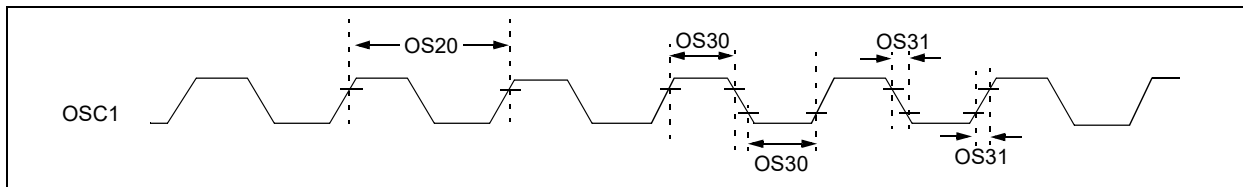
**TABLE 36-14: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS**

| AC CHARACTERISTICS |        |                 | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |            |
|--------------------|--------|-----------------|---|---------------------|------|-------|------------|
| Param. No.         | Symbol | Characteristics | Min.  | Typ. <sup>(1)</sup> | Max. | Units | Conditions |
| DO56               | CL     | All I/O pins    | —   | —                   | 50   | pF    | —          |

**Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-2: EXTERNAL CLOCK TIMING**



**TABLE 36-15: EXTERNAL CLOCK TIMING REQUIREMENTS**

| AC CHARACTERISTICS |               |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                        |                        |       |  |
|--------------------|---------------|--|---|------------------------|------------------------|-------|--|
| Param. No.         | Symbol        | Characteristics  | Minimum   | Typical <sup>(1)</sup> | Maximum                | Units | Conditions   |
| OS10               | Fosc          | External CLKI Frequency<br>(External clocks allowed only<br>in EC and ECPLL modes)                     | DC  | —                      | 64                     | MHz   | EC (Note 2,3)  |
| OS13               |               | Oscillator Crystal Frequency   | 4   | —                      | 32                     | MHz   | HS (Note 2,3)  |
| OS15               |               |  | 32  | 32.768                 | 100                    | kHz   | SOSC crystal<br>ESR must be $\leq 80\text{K ohms}$ (Note 2)              |
| OS20               | Tosc          | $T_{osc} = 1/F_{osc}$  | —   | —                      | —                      | —     | See parameter<br>OS10 for Fosc<br>value                                  |
| OS30               | TosL,<br>TosH | External Clock In (OSC1)<br>High or Low Time   | $0.375 \times T_{osc}$  | —                      | $0.675 \times T_{osc}$ | ns    | EC (Note 2)  |
| OS31               | TosR,<br>TosF | External Clock In (OSC1)<br>Rise or Fall Time  | —   | —                      | 7.5                    | ns    | EC (Note 2)  |
| OS40               | TOST          | Oscillator Start-up Timer Period<br>(Only applies to HS, HSPLL,<br>and SOSC Clock Oscillator<br>modes) | —   | 1024                   | —                      | Tosc  | (Note 2)   |
| OS41               | TFSCM         | Primary Clock Fail Safe<br>Time-out Period   | —   | 2                      | —                      | ms    | (Note 2)   |
| OS42               | GM            | External Oscillator<br>Transconductance  | —   | 16                     | —                      | mAV   | $V_{DD} = 3.3\text{V}$ ,<br>$T_A = +25^{\circ}\text{C}$ , HS<br>(Note 2) |

**Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are characterized but are not tested.

**2:** This parameter is characterized, but not tested in manufacturing.

**3:** See parameter [OS50](#) for PLL input frequency limitations.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-16: SYSTEM PLL TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |       |       |  |
|--------------------|--------|--|---|------|-------|-------|--|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup>                                 | Min.  | Typ. | Max.  | Units | Conditions                                   |
| OS50               | FIN    | PLL Input Frequency Range                                      | 5   | —    | 64    | MHz   | —  |
| OS51               | FSYS   | System Frequency   | DC  | —    | 120   | MHz   | —  |
| OS52               | TLOCK  | PLL Start-up Time (Lock Time)                                  | —   | —    | 100   | μs    | —  |
| OS53               | DCLK   | CLKO Stability <sup>(2)</sup><br>(Period Jitter or Cumulative) | -0.25   | —    | +0.25 | %     | Measured over 100 ms period                  |
| OS54               | FVCO   | PLL Vco Frequency Range  | 350   | —    | 700   | MHz   | FVco output frequency to PLLDIV output       |
| OS54a              | FPLL   | PLL Output Frequency Range                                     | 10  | —    | 120   | MHz   | PLLDIV output frequency range                |
| OS54b              | FPLLI  | VCO Input Frequency Range                                      | 5   | —    | 64    | MHz   | PLLIDIV output frequency range to FVco input |
| OS55a              | FPB    | Peripheral Bus Frequency                                       | DC  | —    | 120   | MHz   | For PBCLKx, 'x' ≠ 6                          |
| OS55b              |        |  | DC  | —    | 30    | MHz   | For PBCLK6                                   |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**Note 2:** This jitter specification is based on clock-cycle by clock-cycle measurements. To get the effective jitter for individual time-bases on communication clocks, use the following formula:

$$EffectiveJitter = \frac{D_{CLK}}{\sqrt{\frac{PBCLKx}{CommunicationClock}}}$$

For example, if PBCLKx = 100 MHz and SPI bit rate = 50 MHz, the effective jitter is as follows:

$$EffectiveJitter = \frac{D_{CLK}}{\sqrt{\frac{100}{50}}} = \frac{D_{CLK}}{1.41}$$

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-17: INTERNAL FRC ACCURACY**

| AC CHARACTERISTICS                              |                 | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |      |       |                     |
|---|-----------------|---|------|------|-------|---------------------|
| Param. No.                                      | Characteristics | Min.  | Typ. | Max. | Units | Conditions          |
| Internal FRC Accuracy @ 8.00 MHz <sup>(1)</sup> |                 |   |      |      |       |                     |
| F20   | FRC             | -4  | —    | +4   | %     | 0°C ≤ TA ≤ +85°C    |
|   |                 | -5  | —    | +5   | %     | -40°C ≤ TA ≤ +125°C |

**Note 1:** Frequency calibrated at 25°C and 3.3V. The TUN bits can be used to compensate for temperature drift.

**TABLE 36-18: INTERNAL LPRC ACCURACY**

| AC CHARACTERISTICS                        |                 | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |      |       |                     |
|---|-----------------|---|------|------|-------|---------------------|
| Param. No.                                | Characteristics | Min.  | Typ. | Max. | Units | Conditions          |
| Internal LPRC @ 32.768 kHz <sup>(1)</sup> |                 |   |      |      |       |                     |
| F21                                       | LPRC            | -8  | —    | +8   | %     | 0°C ≤ TA ≤ +85°C    |
|   |                 | -25   | —    | +25  | %     | -40°C ≤ TA ≤ +125°C |

**Note 1:** Change of LPRC frequency as VDD changes.

**TABLE 36-19: INTERNAL BFRC ACCURACY**

| AC CHARACTERISTICS |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |      |       |                     |
|--------------------|--------------------------------|---|------|------|-------|---------------------|
| Param. No.         | Characteristics <sup>(1)</sup> | Min.  | Typ. | Max. | Units | Comments            |
| BF1                | BFRC                           | -20   | -    | 20   | %     | -40°C ≤ TA ≤ +125°C |

**Note 1:** These parameters are characterized but not tested.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-20: COMPARATOR SPECIFICATIONS**

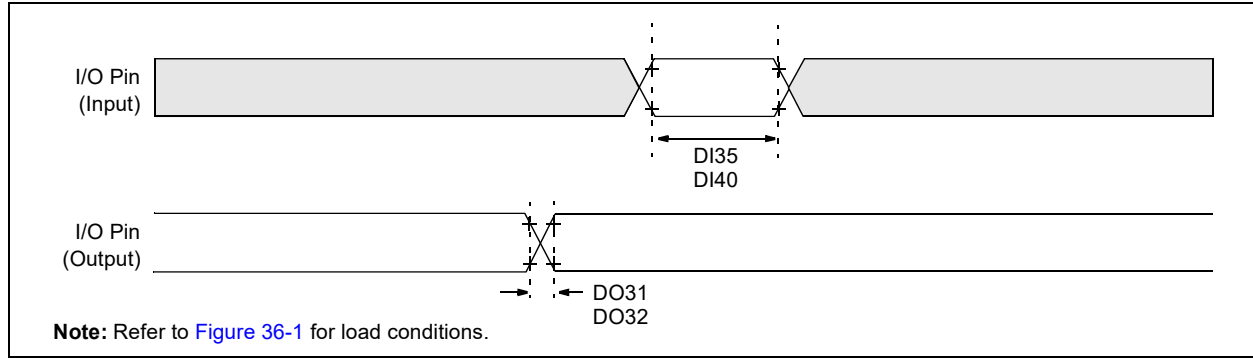
| AC CHARACTERISTICS |          |                                    | Standard Operating Conditions: 3.0V to 3.6V (Note 2)<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |      |       |  |
|--------------------|----------|------------------------------------|--|------|------|-------|--|
| Param. No.         | Symbol   | Characteristics <sup>(1)</sup>     | Min.   | Typ. | Max. | Units | Comments   |
| CM30               | VIOFF    | Input Offset Voltage               | -15  | —    | +15  | mV    | —  |
| CM31               | VICM     | Input Common Mode Voltage          | AVSS   | —    | AVDD | V     | —  |
| CM36a              | VHYST_01 | Input Hysteresis Voltage           | 10   | 15   | 30   | mV    | HYSSEL<1:0> (CMxCON<25:24>) = '0b01 (Lowest)                         |
| CM36b              | VHYST_10 | Input Hysteresis Voltage           | 15   | 30   | 45   | mV    | HYSSEL<1:0> (CMxCON<25:24>) = '0b10                                  |
| CM36c              | VHYST_11 | Input Hysteresis Voltage           | 25   | 45   | 65   | mV    | HYSSEL<1:0> (CMxCON<25:24>) = '0b11 (Highest)                        |
| CM37               | TRESP    | Large Signal Response Time         | —  | —    | 50   | ns    | —  |
| CM38               | TSRESP   | Small Signal Response Time         | —  | —    | 70   | ns    | V <sub>CM</sub> = V <sub>DD</sub> /2; 100 mV step                    |
| CM41               | VIN      | Input Voltage Range                | AVSS   | —    | AVDD | V     | —  |
| CM43               | TON      | Comparator Enabled to Output Valid | —  | 10   | —    | μs    | Comparator module is configured before setting the Comparator ON bit |
| CM44               | TOFF     | Disable to outputs disabled        | —  | 100  | —    | ns    | —  |
| CM47               | CMRR     | Common Mode Rejection              | 39   |      |      |       |  |

**Note 1:** These parameters are characterized but not tested.



# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-3: I/O TIMING CHARACTERISTICS**



**TABLE 36-21: I/O TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended   |      |                     |      |       |               |
|--------------------|--------|---|------|---------------------|------|-------|---------------|
| Param. No.         | Symbol | Characteristics <sup>(2)</sup>  | Min. | Typ. <sup>(1)</sup> | Max. | Units | Conditions    |
| DO31               | TioR   | <b>Port Output Rise Time</b><br>I/O Pins:<br>4x Source Driver Pins -<br>RA0, RA4, RA11, RA12, RA14, RA15,<br>RB0-RB3, RB8, RB9<br>RC0, RC1, RC2, RC10, RC12, RC13<br>RD8, RD12-RD15<br>RE0, RE1, RE8, RE9<br>RF5-RF7, RF9, RF10, RF12, RF13<br>RG0, RG1, RG6-RG15 | —    | —                   | 9.5  | ns    | CLOAD = 50 pF |
|                    |        |   | —    | —                   | 6    | ns    | CLOAD = 20 pF |
|                    |        | <b>Port Output Rise Time</b><br>I/O Pins:<br>8x Source Driver Pins -<br>Replace 8x Source Driver pins with:<br>RA1, RA7, RA8, RA10<br>RB4-RB7, RB10-RB15<br>RC6-RC9, RC11, RC15<br>RD1-RD6<br>RE12-RE15<br>RF0, RF1   | —    | —                   | 8    | ns    | CLOAD = 50 pF |
|                    |        |   | —    | —                   | 6    | ns    | CLOAD = 20 pF |

**Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated.

**Note 2:** This parameter is characterized, but not tested in manufacturing.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-21: I/O TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended   |      |                     |      |       |               |
|--------------------|--------|---|------|---------------------|------|-------|---------------|
| Param. No.         | Symbol | Characteristics <sup>(2)</sup>  | Min. | Typ. <sup>(1)</sup> | Max. | Units | Conditions    |
| DO32               | TioF   | <b>Port Output Fall Time</b><br>I/O Pins:<br>4x Source Driver Pins -<br>RA0, RA4, RA11, RA12, RA14, RA15,<br>RB0-RB3, RB8, RB9<br>RC0, RC1, RC2, RC10, RC12, RC13<br>RD8, RD12-RD15<br>RE0, RE1, RE8, RE9<br>RF5-RF7, RF9, RF10, RF12, RF13<br>RG0, RG1, RG6-RG15 | —    | —                   | 9.5  | ns    | CLOAD = 50 pF |
|                    |        |   | —    | —                   | 6    | ns    | CLOAD = 20 pF |
|                    |        | <b>Port Output Fall Time</b><br>I/O Pins:<br>8x Source Driver Pins -<br>RA1, RA7, RA8, RA10<br>RB4-RB7, RB10-RB15<br>RC6-RC9, RC11, RC15<br>RD1-RD6<br>RE12-RE15<br>RF0, RF1  | —    | —                   | 8    | ns    | CLOAD = 50 pF |
|                    |        |   | —    | —                   | 6    | ns    | CLOAD = 20 pF |
| DI35               | TINP   | INTx Pin High or Low Time   | 5    | —                   | —    | ns    | —             |
| DI40               | TRBP   | CNx High or Low Time (input)  | 5    | —                   | —    | ns    | —             |

**Note 1:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated.

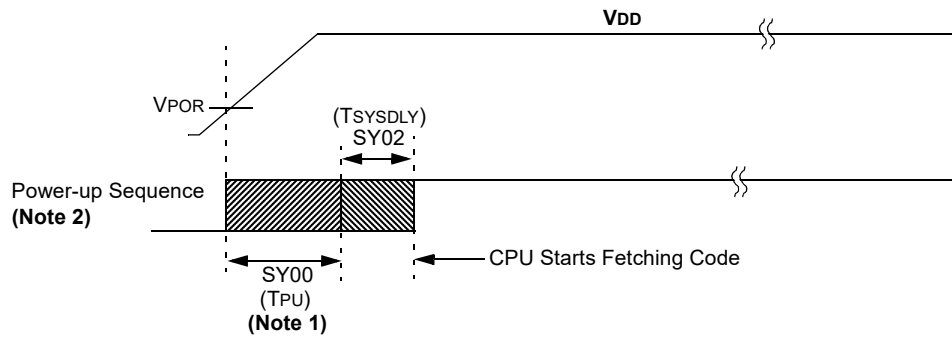
**2:** This parameter is characterized, but not tested in manufacturing.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-4: POWER-ON RESET TIMING CHARACTERISTICS**

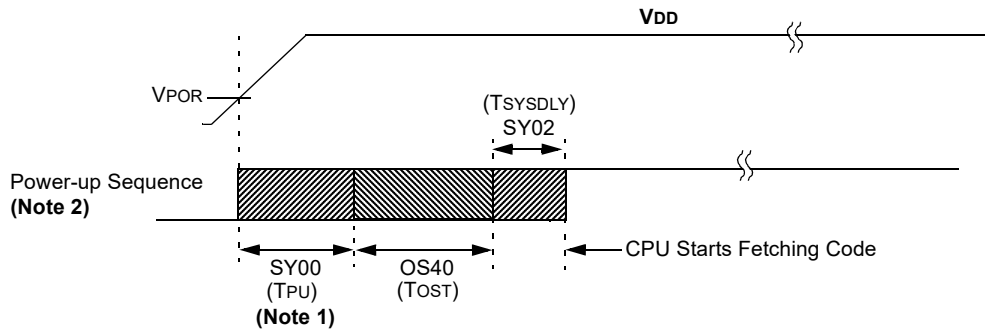
**Internal Voltage Regulator Enabled**

**Clock Sources = (FRC, FRCDIV, FRCDIV16, FRCPLL, EC, ECPLL and LPRC)**



**Internal Voltage Regulator Enabled**

**Clock Sources = (HS, HSPLL, and Sosc)**

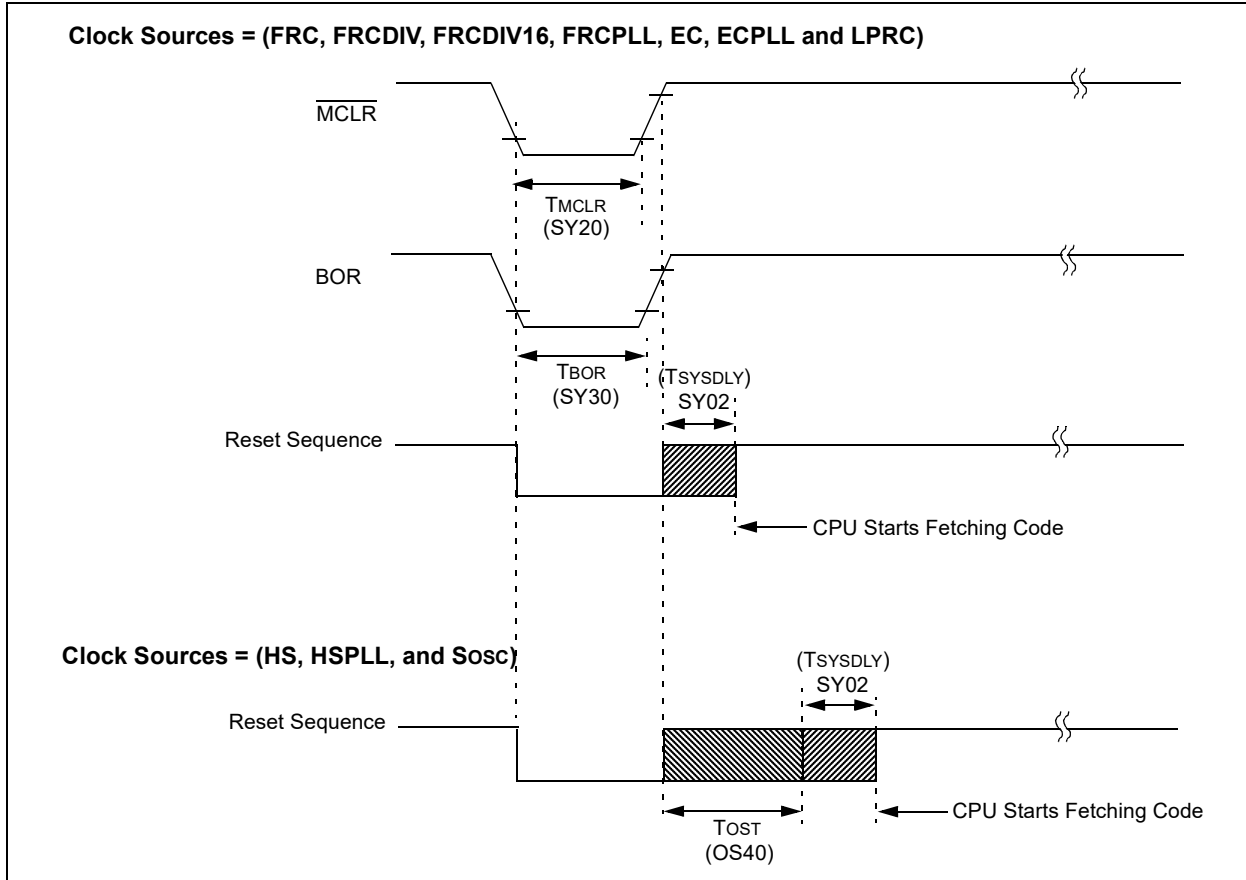


**Note 1:** The power-up period will be extended if the power-up sequence completes before the device exits from BOR ( $V_{DD} < V_{DDMIN}$ ).

**2:** Includes internal voltage regulator stabilization delay.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-5: EXTERNAL RESET TIMING CHARACTERISTICS**



**TABLE 36-22: RESETS TIMING**

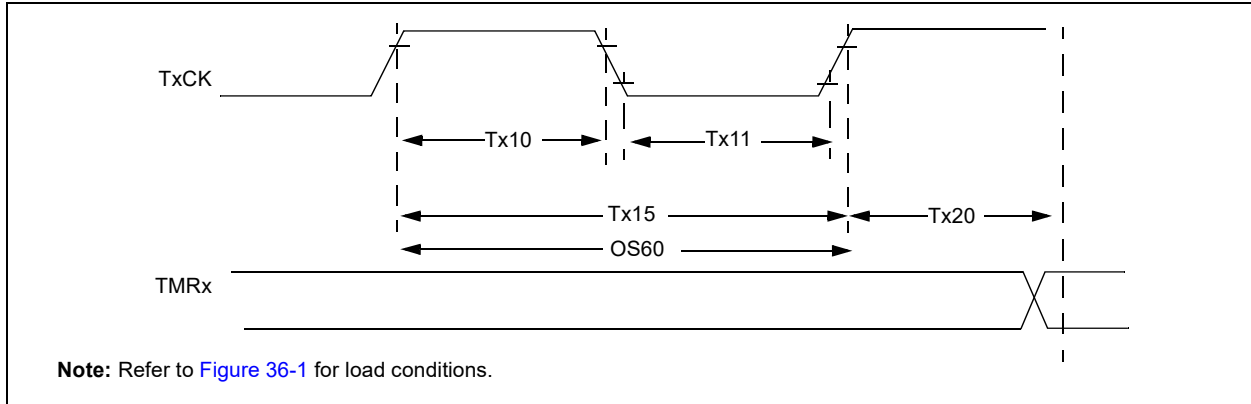
| AC CHARACTERISTICS |         |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |   |      |               |            |
|--------------------|---------|--|---|---|------|---------------|------------|
| Param. No.         | Symbol  | Characteristics <sup>(1)</sup>   | Min.  | Typ. <sup>(2)</sup>                     | Max. | Units         | Conditions |
| SY00               | TPU     | Power-up Period<br>Internal Voltage Regulator Enabled  | —   | 400                                     | R    | $\mu\text{s}$ | —          |
| SY02               | TSYSDLY | System Delay Period:<br>Time Required to Reload Device<br>Configuration Fuses plus SYSCLK<br>Delay before First instruction is<br>Fetched. | —   | $1 \mu\text{s} +$<br>8 SYSCLK<br>cycles | —    | —             | —          |
| SY20               | TMCLR   | MCLR Pulse Width (low)   | 2   | —                                       | —    | $\mu\text{s}$ | —          |
| SY30               | TBOR    | BOR Pulse Width (low)  | —   | 1                                       | —    | $\mu\text{s}$ | —          |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**Note 2:** Data in "Typical" column is at 3.3V, 25°C unless otherwise stated. Characterized by design but not tested.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-6: TIMER1-TIMER9 EXTERNAL CLOCK TIMING CHARACTERISTICS**



**TABLE 36-23: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS<sup>(1)</sup>**

| AC CHARACTERISTICS |           | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                              |  |      |      |         |  |
|--------------------|-----------|---|------------------------------|--|------|------|---------|--|
| Param. No.         | Symbol    | Characteristics <sup>(2)</sup>  |                              | Min.   | Typ. | Max. | Units   | Conditions                             |
| TA10               | TtxH      | TxCK High Time  | Synchronous, with prescaler  | $[(12.5 \text{ ns or } 1 \text{ TPBCLK3}) / N] + 20 \text{ ns}$                  | —    | —    | ns      | Must also meet parameter TA15 (Note 3) |
|                    |           |   | Asynchronous, with prescaler | 10   | —    | —    | ns      |  |
| TA11               | TtxL      | TxCK Low Time   | Synchronous, with prescaler  | $[(12.5 \text{ ns or } 1 \text{ TPBCLK3}) / N] + 20 \text{ ns}$                  | —    | —    | ns      | Must also meet parameter TA15 (Note 3) |
|                    |           |   | Asynchronous, with prescaler | 10   | —    | —    | ns      |  |
| TA15               | TtxP      | TxCK Input Period   | Synchronous, with prescaler  | $[(\text{Greater of } 20 \text{ ns or } 2 \text{ TPBCLK3}) / N] + 30 \text{ ns}$ | —    | —    | ns      | VDD > 2.7V (Note 3)                    |
|                    |           |   | Asynchronous, with prescaler | 20   | —    | —    | ns      | VDD > 2.7V                             |
|                    |           |   | Asynchronous, with prescaler | 50   | —    | —    | ns      | VDD < 2.7V (Note 3)                    |
| OS60               | Ft1       | SOSCO/T1CK Oscillator Input Frequency Range (oscillator enabled by setting TCS bit (T1CON<1>))  |                              | 32   | —    | 50   | kHz     | —                                      |
| TA20               | TCKEXTMRL | Delay from External TxCK Clock Edge to Timer Increment  |                              | —  | —    | 1    | TPBCLK3 | —                                      |

**Note 1:** Timer1 is a Type A.

**Note 2:** This parameter is characterized, but not tested in manufacturing.

**Note 3:** N = Prescale Value (1, 8, 64, 256).

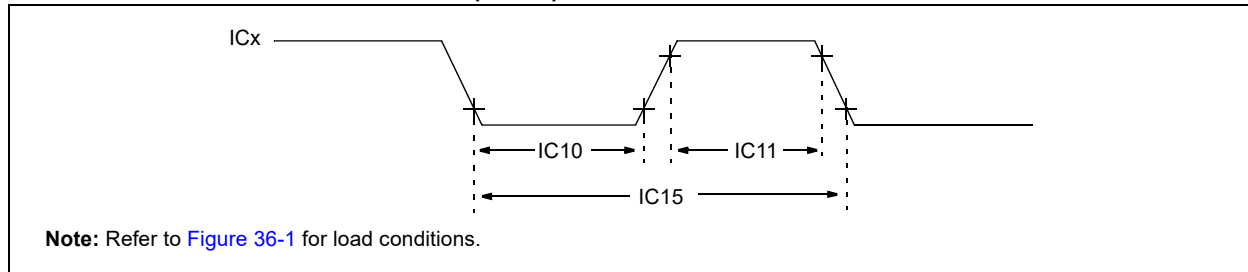
# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-24: TIMER2-TIMER9 EXTERNAL CLOCK TIMING REQUIREMENTS**

| AC CHARACTERISTICS |                       | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                             |   |      |                     |   |
|--------------------|-----------------------|---|-----------------------------|---|------|---------------------|---|
| Param. No.         | Symbol                | Characteristics <sup>(1)</sup>  |                             | Min.  | Max. | Units               | Conditions  |
| TB10               | T <sub>TXH</sub>      | TxCK High Time  | Synchronous, with prescaler | $[(12.5 \text{ ns or } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$                      | —    | ns                  | Must also meet parameter TB15<br>N = prescale value (1, 2, 4, 8, 16, 32, 64, 256) |
| TB11               | T <sub>TXL</sub>      | TxCK Low Time   | Synchronous, with prescaler | $[(12.5 \text{ ns or } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$                      | —    | ns                  |   |
| TB15               | T <sub>TXP</sub>      | TxCK Input Period   | Synchronous, with prescaler | $[(\text{Greater of } [(25 \text{ ns or } 2 \text{ TPBCLK}_3) / N] + 30 \text{ ns})]$ | —    | ns                  |   |
|                    |                       |   |                             | $[(\text{Greater of } [(25 \text{ ns or } 2 \text{ TPBCLK}_3) / N] + 50 \text{ ns})]$ | —    | ns                  | VDD < 2.7V  |
| TB20               | T <sub>CKEXTMRL</sub> | Delay from External TxCK Clock Edge to Timer Increment  |                             | —   | 1    | TPBCLK <sub>3</sub> | —   |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**FIGURE 36-7: INPUT CAPTURE (CAP<sub>x</sub>) TIMING CHARACTERISTICS**



**Note:** Refer to Figure 36-1 for load conditions.

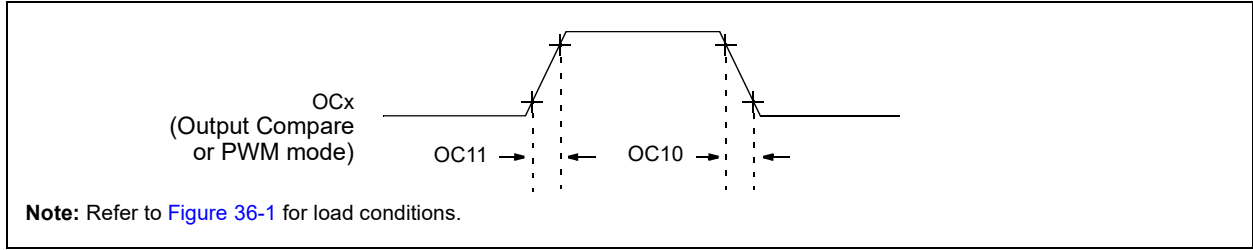
**TABLE 36-25: INPUT CAPTURE MODULE TIMING REQUIREMENTS**

| AC CHARACTERISTICS |                  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |  |   |      |       |   |
|--------------------|------------------|---|--|---|------|-------|---|
| Param. No.         | Symbol           | Characteristics <sup>(1)</sup>  |  | Min.                                      | Max. | Units | Conditions  |
| IC10               | T <sub>cCL</sub> | IC <sub>x</sub> Input Low Time  |  | $((\text{TPBCLK}_x / N) + 25 \text{ ns})$ | —    | ns    | Must also meet parameter IC15.<br>x = 2 for IC1-IC9<br>x = 3 for IC10-IC16<br>N = prescale value (1, 4, 16) |
| IC11               | T <sub>cCH</sub> | IC <sub>x</sub> Input High Time   |  | $((\text{TPBCLK}_x / N) + 25 \text{ ns})$ | —    | ns    |   |
| IC15               | T <sub>cCP</sub> | IC <sub>x</sub> Input Period  |  | $((\text{TPBCLK}_x / N) + 50 \text{ ns})$ | —    | ns    |   |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-8: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS**



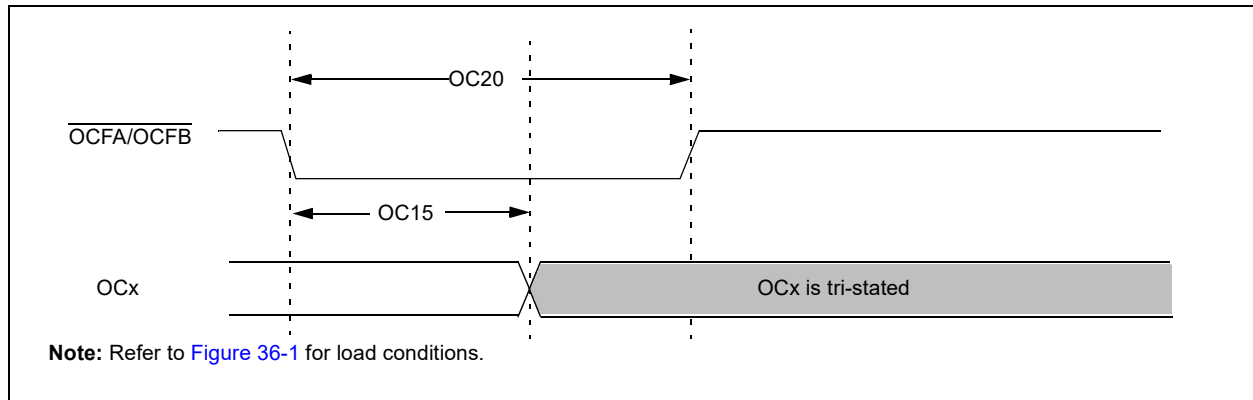
**TABLE 36-26: OUTPUT COMPARE MODULE TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |                    |
|--------------------|--------|--------------------------------|---|---------------------|------|-------|--------------------|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup> | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions         |
| OC10               | TccF   | OCx Output Fall Time           | —   | —                   | —    | ns    | See parameter DO32 |
| OC11               | TccR   | OCx Output Rise Time           | —   | —                   | —    | ns    | See parameter DO31 |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**FIGURE 36-9: OCx/PWM MODULE TIMING CHARACTERISTICS**



**TABLE 36-27: SIMPLE OCx/PWM MODE TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |     |       |            |
|--------------------|--------|--------------------------------|---|---------------------|-----|-------|------------|
| Param No.          | Symbol | Characteristics <sup>(1)</sup> | Min   | Typ. <sup>(2)</sup> | Max | Units | Conditions |
| OC15               | TFD    | Fault Input to PWM I/O Change  | —   | —                   | 50  | ns    | —          |
| OC20               | TFLT   | Fault Input Pulse Width        | 50  | —                   | —   | ns    | —          |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-28: OP AMP SPECIFICATIONS**

| AC CHARACTERISTICS |         |                                 | Standard Operating Conditions: 2.3V to 3.6V (Note 2)<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                       |                       |                                |  |
|--------------------|---------|---------------------------------|--|-----------------------|-----------------------|--------------------------------|--|
| Param. No.         | Symbol  | Characteristics                 | Min.   | Typ. <sup>(1)</sup>   | Max.                  | Units                          | Comments   |
| OA1                | VCMR    | Common Mode Input Voltage Range | AVSS   | —                     | AVDD                  | V                              | —  |
| OA2                | CMRR    | Common Mode Rejection Ratio     | —  | 45                    | —                     | dB                             | VCM = AVDD/2                                       |
| OA3                | VOFFSET | Op amp Offset Voltage           | -5   | —                     | +5                    | mV                             | Non-Unity Gain Mode                                |
| OA5                | ILKG    | Input leakage current           | —  | —                     | See IIL in Table 36-9 | nA                             | —  |
| OA6                | PSRR    | Power Supply Rejection Ratio    | 40   | 52                    | —                     | dB                             | @ 10 kHz   |
|                    |         |                                 | 38   | 49                    | —                     | dB                             | @ 100 kHz  |
|                    |         |                                 | 30   | 42                    | —                     | dB                             | @ 1 MHz  |
| OA7                | TDRIFT  | Amplifier Input Offset Drift    | -15  | —                     | +15                   | $\mu\text{V}/^{\circ}\text{C}$ | With temperature                                   |
| OA8                | VOH     | Amplifier Output Voltage High   | AVSS - 0.300   | —                     | —                     | V                              | ISOURCE $\leq$ 10 mA                               |
|                    |         |                                 | AVSS - 0.1   | —                     | —                     | V                              | ISOURCE $\leq$ 500 $\mu\text{A}$                   |
|                    |         |                                 | AVSS - 0.05  | —                     | —                     | V                              | ISOURCE $\leq$ 200 $\mu\text{A}$                   |
| OA9                | VOL     | Amplifier Output Voltage Low    | —  | —                     | AVDD + 0.300          | V                              | ISINK $\leq$ 10 mA                                 |
|                    |         |                                 | —  | —                     | AVDD + 0.1            | V                              | ISINK $\leq$ 500 $\mu\text{A}$                     |
|                    |         |                                 | —  | —                     | AVDD -+0.05           | V                              | ISINK $\leq$ 200 $\mu\text{A}$                     |
| OA10               | TON     | Enable to Valid Output          | —  | 10                    | —                     | $\mu\text{s}$                  | —  |
| OA11               | TOFF    | Disable to Outputs Disabled     | —  | 100                   | —                     | ns                             | —  |
| OA12               | Ios     | Input Offset Current            | —  | See IIL in Table 36-9 | —                     | —                              | —  |
| OA13               | IB      | Input Bias Current              | —  | See IIL in Table 36-9 | —                     | —                              | —  |
| OA14               | SR      | Slew Rate (Non-Low-power mode)  | 40.0   | —                     | —                     | V/ $\mu\text{s}$               | @ VOUT = 1.0 VPP, VCM = AVDD/2, AVDD = 3.3V        |
| OA14a              | SR      | Slew Rate (Low-power mode)      | 3.7  | —                     | —                     | V/ $\mu\text{s}$               | @ VOUT = 1.0 VPP, VCM = AVDD/2, AVDD = 3.3V        |
| OA15               | GBW     | Gain Bandwidth                  | 48.4   | —                     | —                     | MHz                            | @ VCM = AVDD/2 and AVDD > 2.7V, GAIN $\geq$ 2      |
| OA15A              | GBWLP   | Gain Bandwidth Low-power        | 7.97   | —                     | —                     | MHz                            | @ VCM = AVDD/2 and AVDD > 2.7V, GAIN $\geq$ 2      |
| OA16               | AV      | Gain                            | 2  | —                     | —                     | V/V                            | Minimum op amp stable gain in non-unity gain mode. |
| OA17               | PM      | Phase Margin                    | 34   | —                     | —                     | Degrees                        | —  |

**Note 1:** Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated.

**2:** Device is functional at VBORMIN < VDD < VDDMIN, but will have degraded performance. Device functionality is tested, but not characterized. Analog modules: ADC, op amp/Comparator, and Comparator voltage reference, will have degraded performance. Refer to parameter BO10 in Table 36-5 for the minimum and maximum BOR values.



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-29: UNITY GAIN OP AMP TIMING REQUIREMENTS**

| AC CHARACTERISTICS |         |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |        |                               |
|--------------------|---------|--------------------------------|---|---------------------|------|--------|-------------------------------|
| Param No.          | Symbol  | Characteristics <sup>(2)</sup> | Min.  | Typ. <sup>(1)</sup> | Max. | Units  | Conditions                    |
| UG10               | SR      | Slew Rate                      | 50  | —                   | —    | V/μs   | From 0.5V to 2.5V             |
| UG20               | PM      | Phase Margin                   | 65  | —                   | —    | Degree |                               |
| UG30               | GM      | Gain Margin                    | 5.3   | —                   | —    | dB     |                               |
| UG40               | GBW     | Gain Bandwidth                 | 29  | —                   | —    | MHz    | —                             |
| UG50               | VOFFSET | Opamp Offset Voltage           | -8  | —                   | 8    | mV     |                               |
| UG60               | PSRR    | Power Supply Rejection Ratio   | —   | 59                  | —    | dB     | Specified at 0 Hz             |
| UG70               | PEAK    | Peak Gain                      | —   | —                   | 1.1  | dB     | Gain in excess of 1 (@ 6 MHz) |

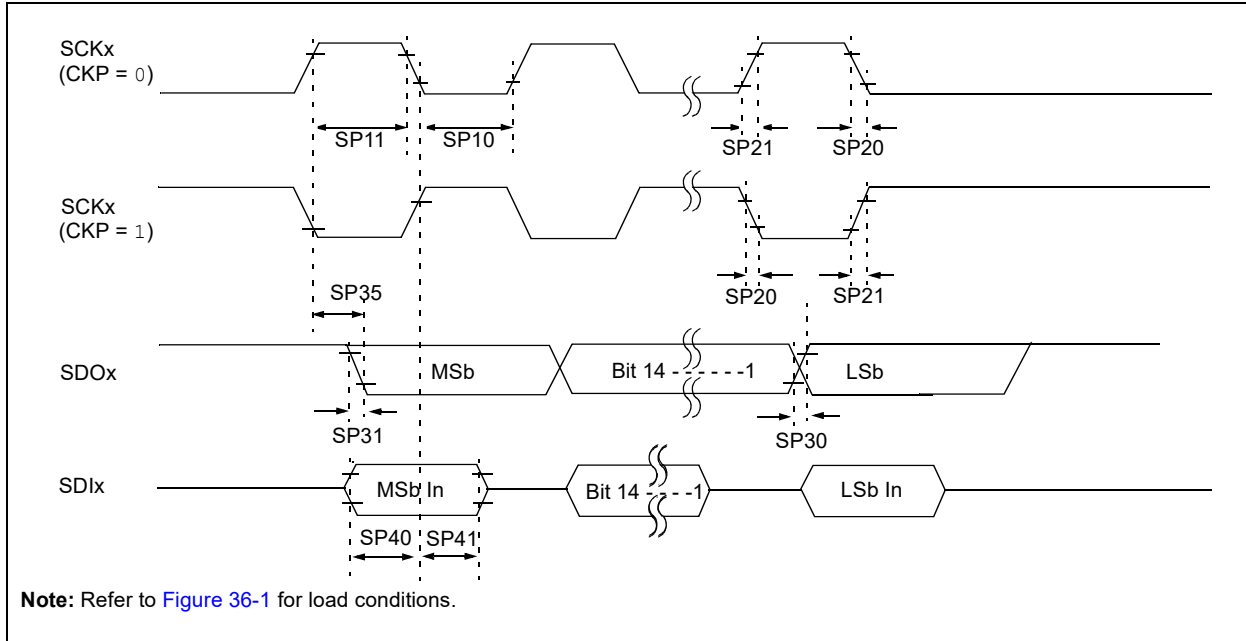
**Note 1:** Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

**2:** All other specifications are identical to the regular op amp mode operation.

**3:** Device is functional at VBORMIN < VDD < VDDMIN, but will have degraded performance. Device functionality is tested, but not characterized. Analog modules: ADC, op amp/Comparator, and Comparator voltage reference, will have degraded performance. Refer to parameter BO10 in Table 36-5 for the minimum and maximum BOR values.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-10: SPIx MODULE MASTER MODE (CKE = 0) TIMING CHARACTERISTICS**



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-30: SPIx MASTER MODE (CKE = 0, SMP = 1) TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |   |
|--------------------|--------|--------------------------------|---|---------------------|------|-------|---|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup> | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions  |
| SP9a               | TSCK   | SCK1 Period (SPI1 only)        | 22.7  | —                   | —    | ns    | (VDD = 3.3V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 35                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 41                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 47                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |

- Note 1:** These parameters are characterized, but not tested in manufacturing.  
**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.  
**3:** Assumes 30 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

TABLE 36-30: SPIx MASTER MODE (CKE = 0, SMP = 1) TIMING REQUIREMENTS (CONTINUED)

| AC CHARACTERISTICS |  |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |  |
|--------------------|--|---|---|---------------------|------|-------|--|
| Param. No.         | Symbol   | Characteristics <sup>(1)</sup>                | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| SP9b               | T <sub>SCK</sub>                               | SCK2 Period (SPI2 only)                       | 35.7  | —                   | —    | ns    | (V <sub>DD</sub> = 3.3V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. |
|                    |  |   | —   | 64                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. |
|                    |  |   | —   | 82                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. |
|                    |  |   | —   | 97                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. |
| SP10               | T <sub>sCL</sub>                               | SCKx Output Low Time                          | T <sub>SCK</sub> /2   | —                   | —    | ns    | —  |
| SP11               | T <sub>sCH</sub>                               | SCKx Output High Time                         | T <sub>SCK</sub> /2   | —                   | —    | ns    | —  |
| SP20               | T <sub>sCF</sub>                               | SCKx Output Fall Time<br>(Note 3)             | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>   |
| SP21               | T <sub>sCR</sub>                               | SCKx Output Rise Time<br>(Note 3)             | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>   |
| SP30               | T <sub>doF</sub>                               | SDOx Data Output Fall Time<br>(Note 3)        | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>   |
| SP31               | T <sub>doR</sub>                               | SDOx Data Output Rise Time<br>(Note 3)        | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>   |
| SP35               | T <sub>sCH2doV</sub> ,<br>T <sub>sCL2doV</sub> | SDOx Data Output Valid after<br>SCKx Edge     | —   | —                   | 7    | ns    | V <sub>DD</sub> > 3.0V   |
|                    |  |   | —   | —                   | 10   | ns    | V <sub>DD</sub> < 3.0V   |
| SP40               | T <sub>diV2sCH</sub> ,<br>T <sub>diV2sCL</sub> | Setup Time of SDIx Data<br>Input to SCKx Edge | 5   | —                   | —    | ns    | —  |
| SP41               | T <sub>sCH2diL</sub> ,<br>T <sub>sCL2diL</sub> | Hold Time of SDIx Data Input<br>to SCKx Edge  | 5   | —                   | —    | ns    | —  |

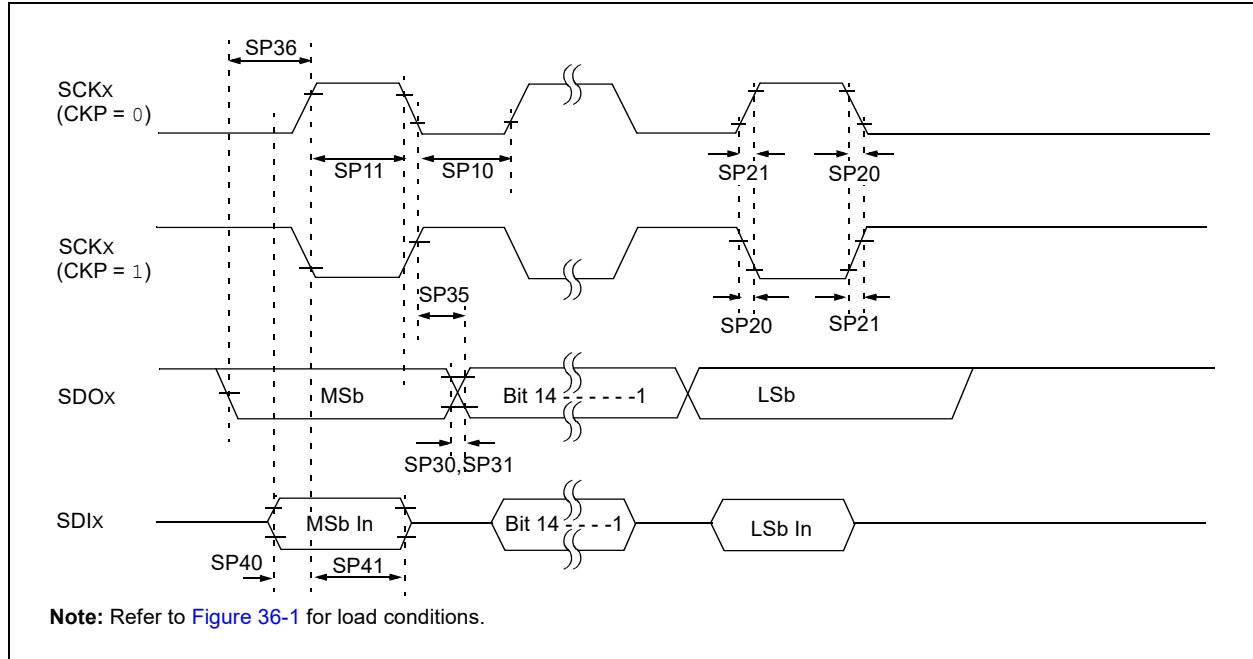
**Note 1:** These parameters are characterized, but not tested in manufacturing.

**Note 2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**Note 3:** Assumes 30 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-11: SPIx MODULE MASTER MODE (CKE = 1) TIMING CHARACTERISTICS**



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-31: SPIx MODULE MASTER MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |  |
|--------------------|--------|--------------------------------|---|---------------------|------|-------|--|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup> | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| SP9a               | TSCK   | SCK1 Period                    | 22.7  | —                   | —    | ns    | (VDD = 3.3V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 27                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 33                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | —   | 39                  | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-31: SPIx MODULE MASTER MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |                  |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |   |
|--------------------|------------------|---|---|---------------------|------|-------|---|
| Param. No.         | Symbol           | Characteristics <sup>(1)</sup>                  | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions  |
| SP9b               | T <sub>SCK</sub> | SCK2 Period                                     | 35.7  | —                   | —    | ns    | (V <sub>DD</sub> = 3.3V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |                  |   | —   | 41                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |                  |   | —   | 59                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |                  |   | —   | 74                  | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIx-CON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. All other remappable SPI pins not contained in conditions for parameter SP9a. |
| SP10               | T <sub>sCL</sub> | SCKx Output Low Time                            | T <sub>SCK</sub> /2   | —                   | —    | ns    | —   |
| SP11               | T <sub>sCH</sub> | SCKx Output High Time                           | T <sub>SCK</sub> /2   | —                   | —    | ns    | —   |
| SP20               | T <sub>sCF</sub> | SCKx Output Fall Time<br>( <b>Note 3</b> )      | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>  |
| SP21               | T <sub>sCR</sub> | SCKx Output Rise Time<br>( <b>Note 3</b> )      | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>  |
| SP30               | T <sub>DOF</sub> | SDOx Data Output Fall Time<br>( <b>Note 3</b> ) | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>  |
| SP30a              | T <sub>SCK</sub> | SCKx Period                                     | 20  | —                   | —    | ns    | Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15.  |
| SP30b              |                  |   | 40  | —                   | —    | ns    | All other remappable SPI pins not contained in conditions for parameter SP9a.   |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-31: SPIx MODULE MASTER MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |                       |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |                                    |
|--------------------|-----------------------|--|---|---------------------|------|-------|------------------------------------|
| Param. No.         | Symbol                | Characteristics <sup>(1)</sup>               | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions                         |
| SP31               | TdoR                  | SDOx Data Output Rise Time ( <b>Note 3</b> ) | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a> |
| SP35               | Tsch2doV,<br>TscL2doV | SDOx Data Output Valid after SCKx Edge       | —   | —                   | 7    | ns    | VDD > 2.7V                         |
|                    |                       |  | —   | —                   | 10   |       | VDD < 2.7V                         |
| SP36               | TdoV2sc,<br>TdoV2scL  | SDOx Data Output Setup to First SCKx Edge    | 7   | —                   | —    | ns    | —                                  |
| SP40               | TdiV2sch,<br>TdiV2scL | Setup Time of SDIx Data Input to SCKx Edge   | 7   | —                   | —    | ns    | VDD > 2.7V                         |
|                    |                       |  | 10  |                     |      |       | VDD < 2.7V                         |
| SP41               | Tsch2diL,<br>TscL2diL | Hold Time of SDIx Data Input to SCKx Edge    | 7   | —                   | —    | ns    | VDD > 2.7V                         |
|                    |                       |  | 10  | —                   | —    | ns    | VDD < 2.7V                         |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

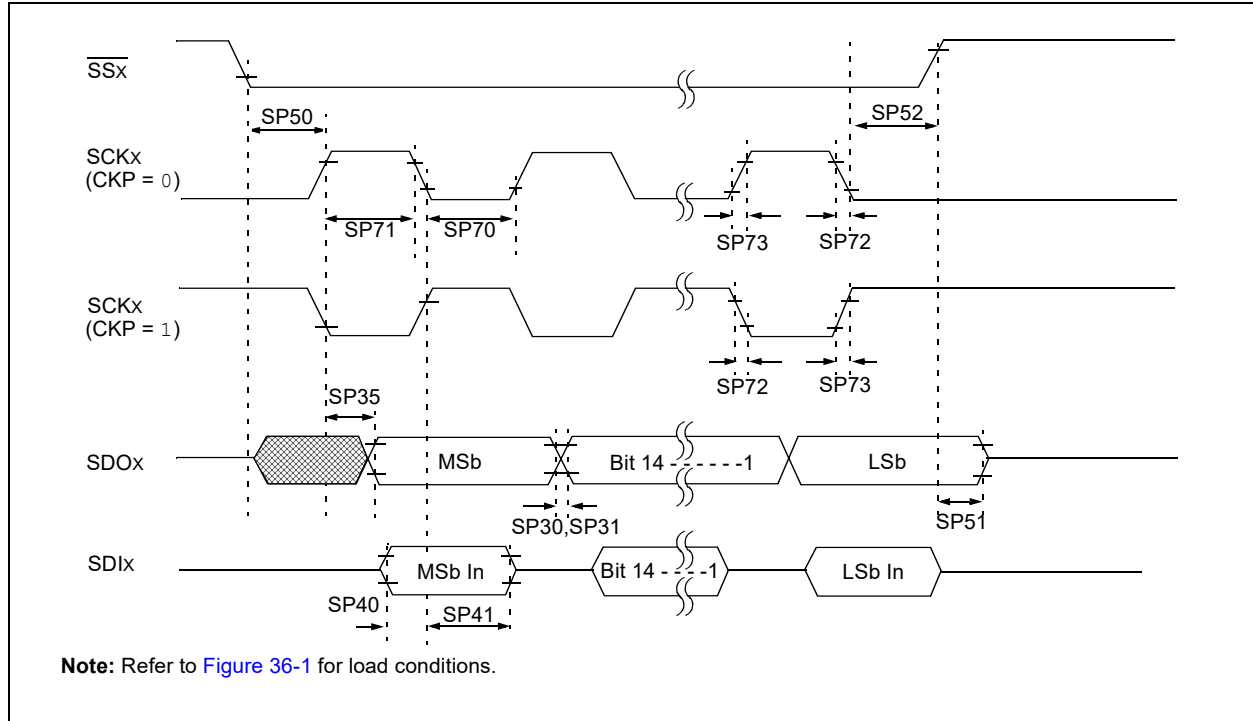
**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.



# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-12: SPIx MODULE SLAVE MODE (CKE = 0) TIMING CHARACTERISTICS**



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-32: SPIx MODULE SLAVE MODE (CKE = 0, SMP = 1) TIMING REQUIREMENTS**

| AC CHARACTERISTICS |                  |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |  |
|--------------------|------------------|--------------------------------|---|---------------------|------|-------|--|
| Param. No.         | Symbol           | Characteristics <sup>(1)</sup> | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| SP9a               | T <sub>SCK</sub> | SCKx Period                    | 20  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0)<br>Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |                  |                                | 27  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0)<br>Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |                  |                                | 33  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1)<br>Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |                  |                                | 39  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1)<br>Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-32: SPIx MODULE SLAVE MODE (CKE = 0, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |  |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |  |
|--------------------|--|---|---|---------------------|------|-------|--|
| Param. No.         | Symbol   | Characteristics <sup>(1)</sup>                | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| SP9b               | T <sub>SCK</sub>                               | SCKx Period                                   | 22  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0<br>All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |   | 41  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0<br>All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |   | 59  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1<br>All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |   | 74  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1<br>All other remappable SPI pins not contained in conditions for parameter SP9a. |
| SP70               | T <sub>sCL</sub>                               | SCKx Input Low Time                           | T <sub>SCK</sub> /2   | —                   | —    | ns    | —  |
| SP71               | T <sub>sCH</sub>                               | SCKx Input High Time                          | T <sub>SCK</sub> /2   | —                   | —    | ns    | —  |
| SP72               | T <sub>sCF</sub>                               | SCKx Input Fall Time                          | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>   |
| SP73               | T <sub>sCR</sub>                               | SCKx Input Rise Time                          | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>   |
| SP30               | T <sub>doF</sub>                               | SDOx Data Output Fall Time<br><b>(Note 3)</b> | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>   |
| SP31               | T <sub>doR</sub>                               | SDOx Data Output Rise Time<br><b>(Note 3)</b> | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>   |
| SP35               | T <sub>sCH2doV</sub> ,<br>T <sub>sCL2doV</sub> | SDOx Data Output Valid after SCKx Edge        | —   | —                   | 7    | ns    | V <sub>DD</sub> > 2.7V   |
|                    |  |   | —   | —                   | 10   | ns    | V <sub>DD</sub> < 2.7V   |
| SP40               | T <sub>dIV2sCH</sub> ,<br>T <sub>dIV2sCL</sub> | Setup Time of SDIx Data Input to SCKx Edge    | 5   | —                   | —    | ns    | —  |
| SP41               | T <sub>sCH2dIL</sub> ,<br>T <sub>sCL2dIL</sub> | Hold Time of SDIx Data Input to SCKx Edge     | 5   | —                   | —    | ns    | —  |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

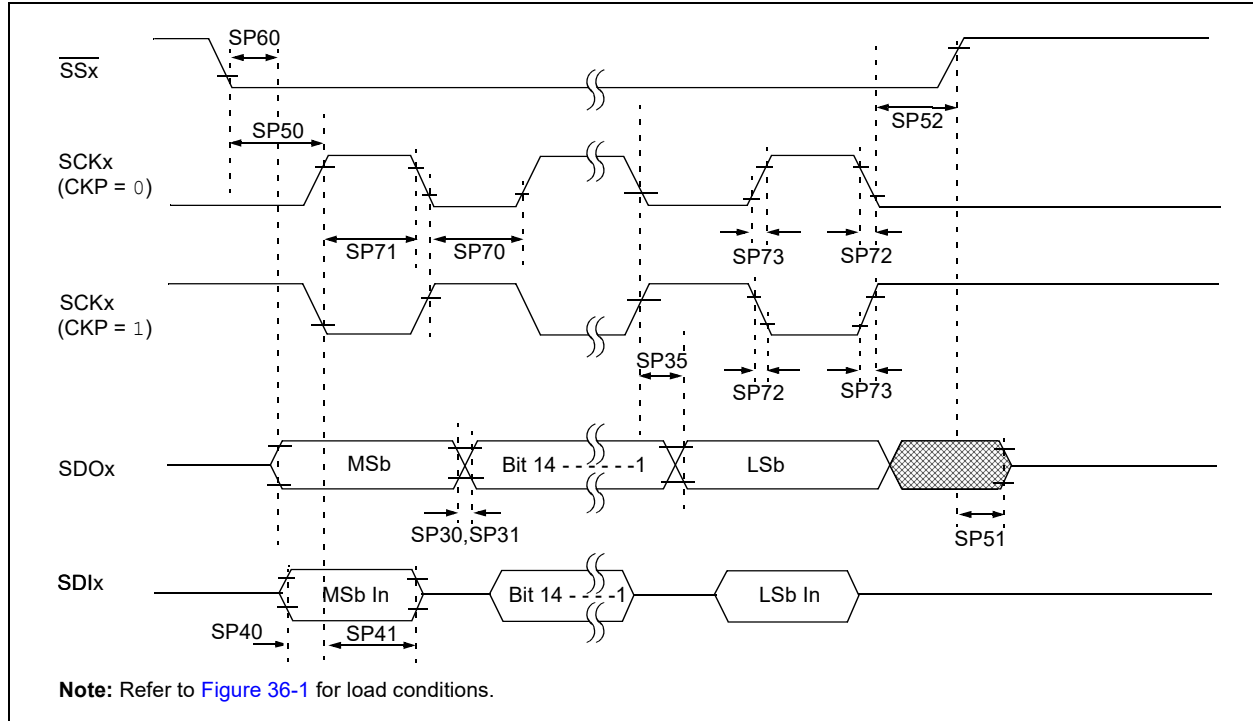
**TABLE 36-32: SPIx MODULE SLAVE MODE (CKE = 0, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |                       |                                     | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |            |
|--------------------|-----------------------|-------------------------------------|---|---------------------|------|-------|------------|
| Param. No.         | Symbol                | Characteristics <sup>(1)</sup>      | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions |
| SP50               | TssL2sch,<br>TssL2scl | SSx ↓ to SCKx ↑ or SCKx Input       | 88  | —                   | —    | ns    | —          |
| SP51               | TssH2doz              | SSx ↑ to SDOx Output High-Impedance | 2.5   | —                   | 12   | ns    | —          |
| SP52               | Tsch2ssh<br>TscL2ssh  | SSx after SCKx Edge                 | 10  | —                   | —    | ns    | —          |

- Note 1:** These parameters are characterized, but not tested in manufacturing.  
**Note 2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.  
**Note 3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-13: SPIx MODULE SLAVE MODE (CKE = 1) TIMING CHARACTERISTICS**



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-33: SPIx MODULE SLAVE MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |   |
|--------------------|--------|--------------------------------|---|---------------------|------|-------|---|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup> | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions  |
| SP9a               | Tsck   | SCKx Period                    | 20  | —                   | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | 27  | —                   | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | 33  | —                   | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |
|                    |        |                                | 39  | —                   | —    | ns    | (VDD ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. Dedicated SCK1 and SCK2 on RB7 and RB6, respectively or PPS remappable SPI onto pins RB5, RA1, and RB15. |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-33: SPIx MODULE SLAVE MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |  |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |  |
|--------------------|--|--|---|---------------------|------|-------|--|
| Param. No.         | Symbol   | Characteristics <sup>(1)</sup>               | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| SP9b               | T <sub>SCK</sub>                               | SCKx Period                                  | 22  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 0. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |  | 41  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 0. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |  | 59  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 0, SRCON1x.y = 1. All other remappable SPI pins not contained in conditions for parameter SP9a. |
|                    |  |  | 74  | —                   | —    | ns    | (V <sub>DD</sub> ≥ 3.0V and the SMP bit (SPIxCON<9> = 1), I/O Pin Slew Rate Control (x = A-F, y = port pin), SRCON0x.y = 1, SRCON1x.y = 1. All other remappable SPI pins not contained in conditions for parameter SP9a. |
| SP70               | T <sub>sCL</sub>                               | SCKx Input Low Time                          | T <sub>sCK</sub> /2   | —                   | —    | ns    | —  |
| SP71               | T <sub>sCH</sub>                               | SCKx Input High Time                         | T <sub>sCK</sub> /2   | —                   | —    | ns    | —  |
| SP72               | T <sub>sCF</sub>                               | SCKx Input Fall Time                         | —   | —                   | 10   | ns    | —  |
| SP73               | T <sub>sCR</sub>                               | SCKx Input Rise Time                         | —   | —                   | 10   | ns    | —  |
| SP30               | T <sub>doF</sub>                               | SDOx Data Output Fall Time ( <b>Note 3</b> ) | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a>   |
| SP31               | T <sub>doR</sub>                               | SDOx Data Output Rise Time ( <b>Note 3</b> ) | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a>   |
| SP35               | T <sub>sCH2doV</sub> ,<br>T <sub>sCL2doV</sub> | SDOx Data Output Valid after SCKx Edge       | —   | —                   | 10   | ns    | V <sub>DD</sub> > 2.7V   |
|                    |  |  | —   | —                   | 15   | ns    | V <sub>DD</sub> < 2.7V   |
| SP40               | T <sub>diV2sCH</sub> ,<br>T <sub>diV2sCL</sub> | Setup Time of SDIx Data Input to SCKx Edge   | 0   | —                   | —    | ns    | —  |
| SP41               | T <sub>sCH2dIL</sub> ,<br>T <sub>sCL2dIL</sub> | Hold Time of SDIx Data Input to SCKx Edge    | 7   | —                   | —    | ns    | —  |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** Assumes 10 pF load on all SPIx pins.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-33: SPIx MODULE SLAVE MODE (CKE = 1, SMP = 1) TIMING REQUIREMENTS (CONTINUED)**

| AC CHARACTERISTICS |                       |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |      |       |            |
|--------------------|-----------------------|---|---|---------------------|------|-------|------------|
| Param. No.         | Symbol                | Characteristics <sup>(1)</sup>  | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions |
| SP50               | TssL2sch,<br>TssL2scl | $\overline{\text{SS}}_x \downarrow$ to SCKx $\downarrow$ or SCKx $\uparrow$ Input | 88  | —                   | —    | ns    | —          |
| SP51               | TssH2doZ              | $\overline{\text{SS}}_x \uparrow$ to SDOx Output High-Impedance ( <b>Note 3</b> ) | 2.5   | —                   | 12   | ns    | —          |
| SP52               | Tsch2ssH<br>TscL2ssH  | $\overline{\text{SS}}_x \uparrow$ after SCKx Edge                                 | 10  | —                   | —    | ns    | —          |
| SP60               | TssL2doV              | SDOx Data Output Valid after $\overline{\text{SS}}_x$ Edge                        | —   | —                   | 12.5 | ns    | —          |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

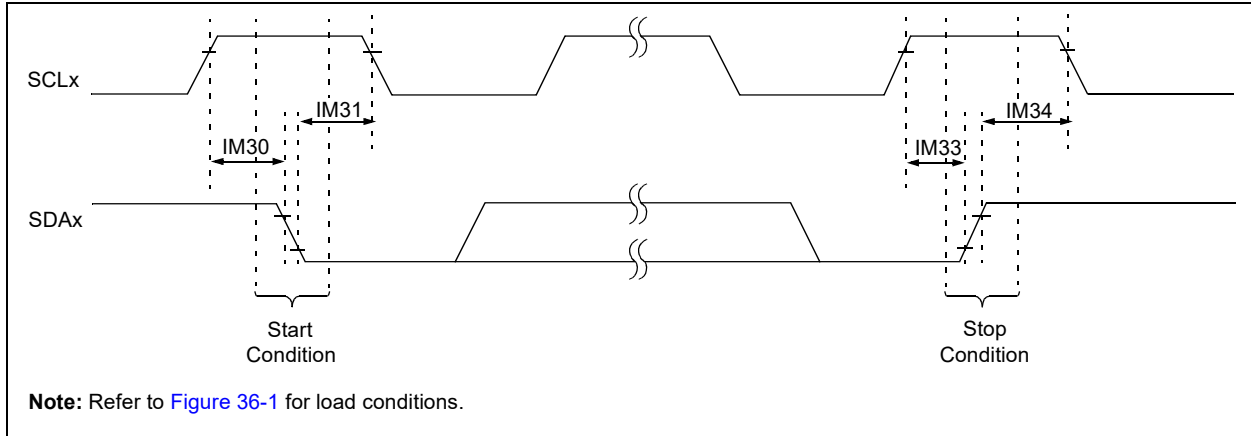
**Note 2:** Data in the “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**Note 3:** Assumes 10 pF load on all SPIx pins.

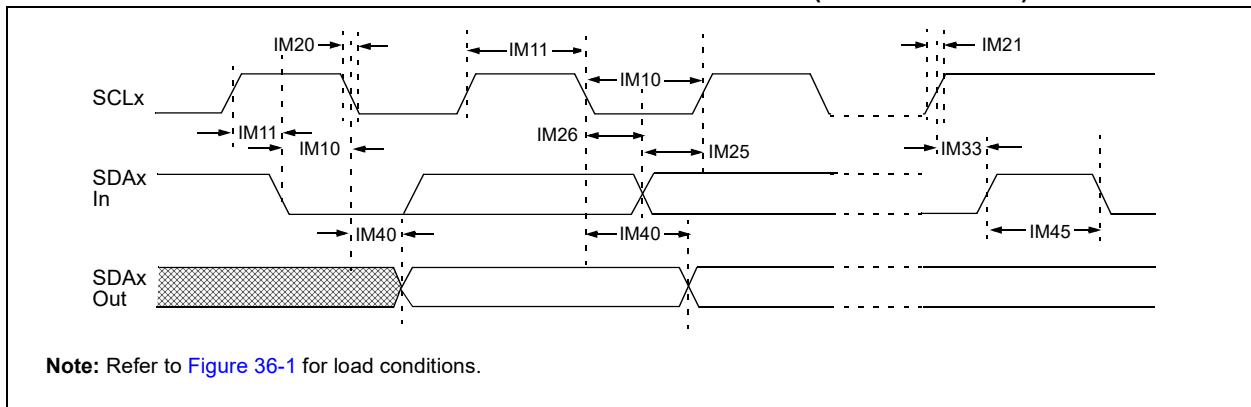


# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-14: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)**



**FIGURE 36-15: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)**



**TABLE 36-34: I2CX BUS DATA TIMING REQUIREMENTS (MASTER MODE)**

| AC CHARACTERISTICS |         | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +105°C for V-temp |                               |                 |       |            |   |
|--------------------|---------|--|-------------------------------|-----------------|-------|------------|---|
| Param. No.         | Symbol  | Characteristics  | Min. <sup>(1)</sup>           | Max.            | Units | Conditions |   |
| IM10               | TLO:SCL | Clock Low Time   | 100 kHz mode                  | TPB * (BRG + 2) | —     | μs         | — |
|                    |         |  | 400 kHz mode                  | TPB * (BRG + 2) | —     | μs         | — |
|                    |         |  | 1 MHz mode<br><b>(Note 2)</b> | TPB * (BRG + 2) | —     | μs         | — |
| IM11               | THI:SCL | Clock High Time  | 100 kHz mode                  | TPB * (BRG + 2) | —     | μs         | — |
|                    |         |  | 400 kHz mode                  | TPB * (BRG + 2) | —     | μs         | — |
|                    |         |  | 1 MHz mode<br><b>(Note 2)</b> | TPB * (BRG + 2) | —     | μs         | — |

**Note 1:** BRG is the value of the I<sup>2</sup>C Baud Rate Generator.

**2:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

**3:** The typical value for this parameter is 104 ns.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-34: I2CX BUS DATA TIMING REQUIREMENTS (MASTER MODE) (CONTINUED)**

| AC CHARACTERISTICS |         |                               |                        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +105°C for V-temp |       |            |   |
|--------------------|---------|-------------------------------|------------------------|--|-------|------------|---|
| Param. No.         | Symbol  | Characteristics               | Min. <sup>(1)</sup>    | Max.   | Units | Conditions |   |
| IM20               | TF:SCL  | SDAx and SCLx<br>Fall Time    | 100 kHz mode           | —  | 300   | ns         | CB is specified to be<br>from 10 to 200 pF                  |
|                    |         |                               | 400 kHz mode           | 20 + 0.1 CB  | 300   | ns         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | —  | 100   | ns         |   |
| IM21               | TR:SCL  | SDAx and SCLx<br>Rise Time    | 100 kHz mode           | —  | 1000  | ns         | CB is specified to be<br>from 10 to 200 pF                  |
|                    |         |                               | 400 kHz mode           | 20 + 0.1 CB  | 300   | ns         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | —  | 300   | ns         |   |
| IM25               | TSU:DAT | Data Input<br>Setup Time      | 100 kHz mode           | 250  | —     | ns         | —   |
|                    |         |                               | 400 kHz mode           | 100  | —     | ns         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | 100  | —     | ns         |   |
| IM26               | THD:DAT | Data Input<br>Hold Time       | 100 kHz mode           | 0  | —     | μs         | —   |
|                    |         |                               | 400 kHz mode           | 0  | 0.9   | μs         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | 0  | 0.3   | μs         |   |
| IM30               | TSU:STA | Start Condition<br>Setup Time | 100 kHz mode           | TPB * (BRG + 2)  | —     | μs         | Only relevant for<br>Repeated Start<br>condition            |
|                    |         |                               | 400 kHz mode           | TPB * (BRG + 2)  | —     | μs         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | TPB * (BRG + 2)  | —     | μs         |   |
| IM31               | THD:STA | Start Condition<br>Hold Time  | 100 kHz mode           | TPB * (BRG + 2)  | —     | μs         | After this period, the<br>first clock pulse is<br>generated |
|                    |         |                               | 400 kHz mode           | TPB * (BRG + 2)  | —     | μs         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | TPB * (BRG + 2)  | —     | μs         |   |
| IM33               | TSU:STO | Stop Condition<br>Setup Time  | 100 kHz mode           | TPB * (BRG + 2)  | —     | μs         | —   |
|                    |         |                               | 400 kHz mode           | TPB * (BRG + 2)  | —     | μs         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | TPB * (BRG + 2)  | —     | μs         |   |
| IM34               | THD:STO | Stop Condition<br>Hold Time   | 100 kHz mode           | TPB * (BRG + 2)  | —     | ns         | —   |
|                    |         |                               | 400 kHz mode           | TPB * (BRG + 2)  | —     | ns         |   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | TPB * (BRG + 2)  | —     | ns         |   |
| IM40               | TAA:SCL | Output Valid<br>from Clock    | 100 kHz mode           | —  | 3500  | ns         | —   |
|                    |         |                               | 400 kHz mode           | —  | 1000  | ns         | —   |
|                    |         |                               | 1 MHz mode<br>(Note 2) | —  | 350   | ns         | —   |

**Note 1:** BRG is the value of the I<sup>2</sup>C Baud Rate Generator.

**2:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

**3:** The typical value for this parameter is 104 ns.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-34: I2CX BUS DATA TIMING REQUIREMENTS (MASTER MODE) (CONTINUED)**

| AC CHARACTERISTICS |         |                        |                        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +105°C for V-temp |      |       |   |
|--------------------|---------|------------------------|------------------------|--|------|-------|---|
| Param. No.         | Symbol  | Characteristics        |                        | Min. <sup>(1)</sup>  | Max. | Units | Conditions  |
| IM45               | TBF:SDA | Bus Free Time          | 100 kHz mode           | 4.7  | —    | μs    | The amount of time the bus must be free before a new transmission can start |
|                    |         |                        | 400 kHz mode           | 1.3  | —    | μs    |   |
|                    |         |                        | 1 MHz mode<br>(Note 2) | 0.5  | —    | μs    |   |
| IM50               | CB      | Bus Capacitive Loading |                        | —  | 200  | pF    | —   |
| IM51               | TPGD    | Pulse Gobbler Delay    |                        | 52   | 312  | ns    | See Note 3  |

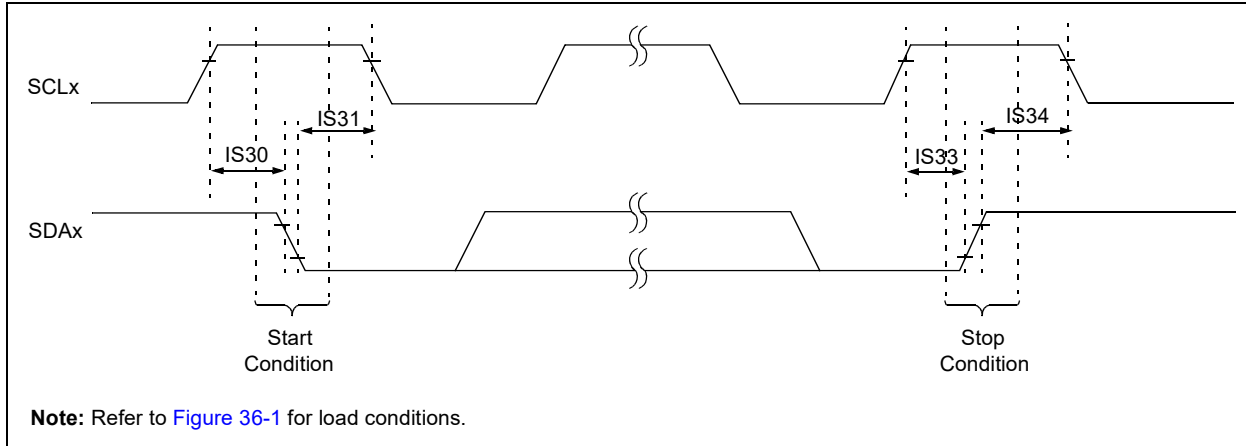
**Note 1:** BRG is the value of the I<sup>2</sup>C Baud Rate Generator.

**2:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

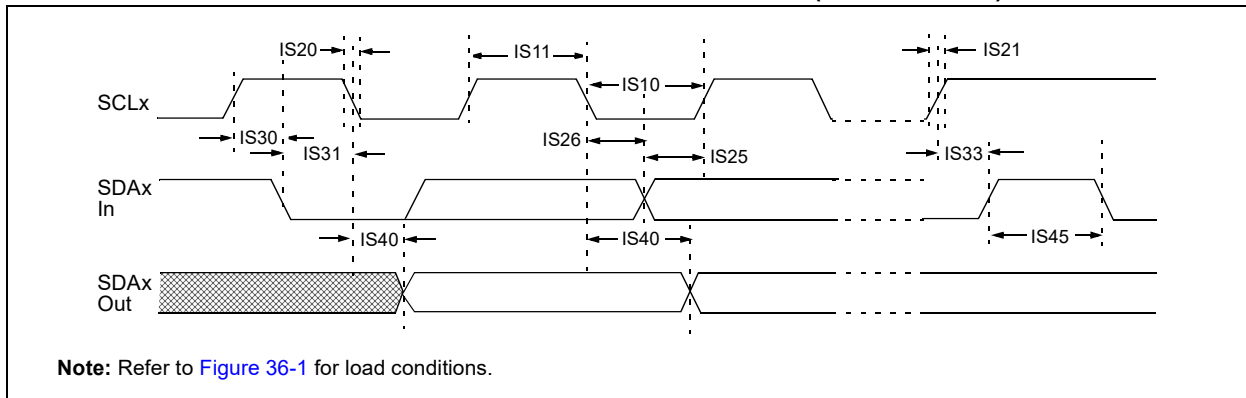
**3:** The typical value for this parameter is 104 ns.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-16: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)**



**FIGURE 36-17: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)**



**TABLE 36-35: I2CX BUS DATA TIMING REQUIREMENTS (SLAVE MODE)**

| AC CHARACTERISTICS |         |                 |                        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +105°C for V-temp |      |       |  |
|--------------------|---------|-----------------|------------------------|--|------|-------|--|
| Param. No.         | Symbol  | Characteristics |                        | Min.   | Max. | Units | Conditions                                 |
| IS10               | TLO:SCL | Clock Low Time  | 100 kHz mode           | 4.7  | —    | μs    | PBCLK must operate at a minimum of 800 kHz |
|                    |         |                 | 400 kHz mode           | 1.3  | —    | μs    | PBCLK must operate at a minimum of 3.2 MHz |
|                    |         |                 | 1 MHz mode<br>(Note 1) | 0.5  | —    | μs    | —  |
| IS11               | THI:SCL | Clock High Time | 100 kHz mode           | 4.0  | —    | μs    | PBCLK must operate at a minimum of 800 kHz |
|                    |         |                 | 400 kHz mode           | 0.6  | —    | μs    | PBCLK must operate at a minimum of 3.2 MHz |
|                    |         |                 | 1 MHz mode<br>(Note 1) | 0.5  | —    | μs    | —  |

**Note 1:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

# PIC32MK GPG/MCJ with CAN FD Family

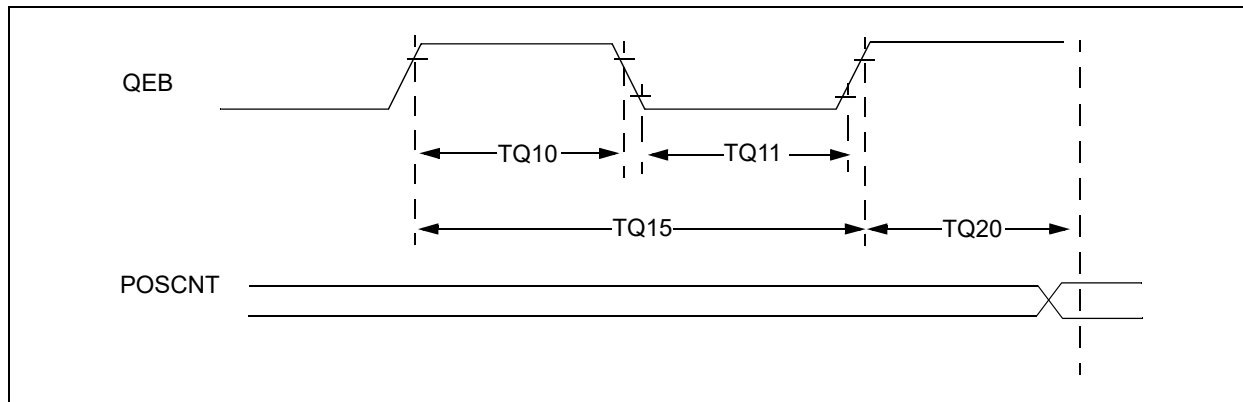
**TABLE 36-35: I2CX BUS DATA TIMING REQUIREMENTS (SLAVE MODE) (CONTINUED)**

| AC CHARACTERISTICS |         |                               |                        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +105°C for V-temp |      |       |   |
|--------------------|---------|-------------------------------|------------------------|--|------|-------|---|
| Param. No.         | Symbol  | Characteristics               |                        | Min.   | Max. | Units | Conditions  |
| IS20               | TF:SCL  | SDAx and SCLx<br>Fall Time    | 100 kHz mode           | —  | 300  | ns    | CB is specified to be from<br>10 to 200 pF  |
|                    |         |                               | 400 kHz mode           | 20 + 0.1 CB  | 300  | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | —  | 100  | ns    |   |
| IS21               | TR:SCL  | SDAx and SCLx<br>Rise Time    | 100 kHz mode           | —  | 1000 | ns    | CB is specified to be from<br>10 to 200 pF  |
|                    |         |                               | 400 kHz mode           | 20 + 0.1 CB  | 300  | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | —  | 300  | ns    |   |
| IS25               | TSU:DAT | Data Input<br>Setup Time      | 100 kHz mode           | 250  | —    | ns    | —   |
|                    |         |                               | 400 kHz mode           | 100  | —    | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 100  | —    | ns    |   |
| IS26               | THD:DAT | Data Input<br>Hold Time       | 100 kHz mode           | 0  | —    | ns    | —   |
|                    |         |                               | 400 kHz mode           | 0  | 0.9  | μs    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 0  | 0.3  | μs    |   |
| IS30               | TSU:STA | Start Condition<br>Setup Time | 100 kHz mode           | 4700   | —    | ns    | Only relevant for Repeated<br>Start condition                                     |
|                    |         |                               | 400 kHz mode           | 600  | —    | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 250  | —    | ns    |   |
| IS31               | THD:STA | Start Condition<br>Hold Time  | 100 kHz mode           | 4000   | —    | ns    | After this period, the first<br>clock pulse is generated                          |
|                    |         |                               | 400 kHz mode           | 600  | —    | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 250  | —    | ns    |   |
| IS33               | TSU:STO | Stop Condition<br>Setup Time  | 100 kHz mode           | 4000   | —    | ns    | —   |
|                    |         |                               | 400 kHz mode           | 600  | —    | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 600  | —    | ns    |   |
| IS34               | THD:STO | Stop Condition<br>Hold Time   | 100 kHz mode           | 4000   | —    | ns    | —   |
|                    |         |                               | 400 kHz mode           | 600  | —    | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 250  | —    | ns    |   |
| IS40               | TAA:SCL | Output Valid from<br>Clock    | 100 kHz mode           | 0  | 3500 | ns    | —   |
|                    |         |                               | 400 kHz mode           | 0  | 1000 | ns    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 0  | 350  | ns    |   |
| IS45               | TBF:SDA | Bus Free Time                 | 100 kHz mode           | 4.7  | —    | μs    | The amount of time the bus<br>must be free before a new<br>transmission can start |
|                    |         |                               | 400 kHz mode           | 1.3  | —    | μs    |   |
|                    |         |                               | 1 MHz mode<br>(Note 1) | 0.5  | —    | μs    |   |
| IS50               | CB      | Bus Capacitive Loading        |                        | —  | 400  | pF    | —   |

**Note 1:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-18: QEI MODULE EXTERNAL CLOCK TIMING CHARACTERISTICS**



**TABLE 36-36: QEI MODULE EXTERNAL CLOCK TIMING REQUIREMENTS**

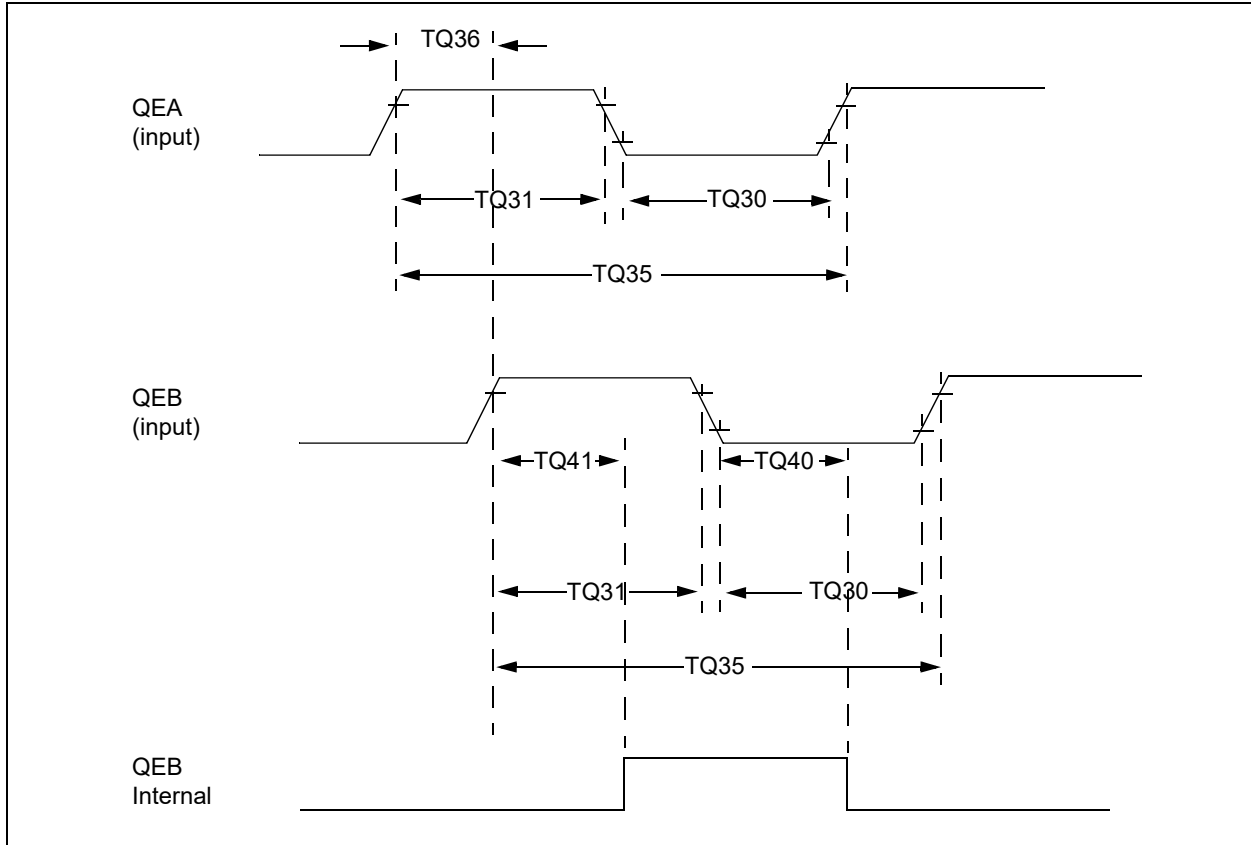
| AC CHARACTERISTICS |           | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                             |  |      |                 |       |  |
|--------------------|-----------|---|-----------------------------|--|------|-----------------|-------|--|
| Param No.          | Symbol    | Characteristic <sup>(1)</sup>   |                             | Min.                                       | Typ. | Max.            | Units | Conditions   |
| TQ10               | TtQH      | TQCK High Time  | Synchronous, with prescaler | $[(12.5 \text{ or } 0.5 T_{CY}) / N] + 25$ | —    | —               | ns    | Must also meet parameter TQ15.<br><br>N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 2</b> ) |
| TQ11               | TtQL      | TQCK Low Time   | Synchronous, with prescaler | $[(12.5 \text{ or } 0.5 T_{CY}) / N] + 25$ | —    | —               | ns    | Must also meet parameter TQ15.<br><br>N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 2</b> ) |
| TQ15               | TtQP      | TQCP Input Period   | Synchronous, with prescaler | $[(25 \text{ or } T_{CY}) / N] + 50$       | —    | —               | ns    | N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 2</b> )                                       |
| TQ20               | TCKEXTMRL | Delay from External TxCK Clock Edge to Timer Increment  |                             | —  | 1    | T <sub>CY</sub> | —     | —  |

**Note 1:** These parameters are characterized but not tested in manufacturing.

**Note 2:** N = Index Channel Digital Filter Clock Divide Select bits.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-19: QEA/QEB INPUT CHARACTERISTICS**



**TABLE 36-37: QUADRATURE DECODER TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |  |
|--------------------|--------|---|---------------------|------|-------|--|
| Param No.          | Symbol | Characteristic <sup>(1)</sup>   | Typ. <sup>(2)</sup> | Max. | Units | Conditions   |
| TQ30               | TQuL   | Quadrature Input Low Time   | 6 TCY               | —    | ns    | —  |
| TQ31               | TQuH   | Quadrature Input High Time  | 6 TCY               | —    | ns    | —  |
| TQ35               | TQuIN  | Quadrature Input Period   | 12 TCY              | —    | ns    | —  |
| TQ36               | TQuP   | Quadrature Phase Period   | 3 TCY               | —    | ns    | —  |
| TQ40               | TQuFL  | Filter Time to Recognize Low, with Digital Filter   | 3 * N * TCY         | —    | ns    | N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 3</b> ) |
| TQ41               | TQuFH  | Filter Time to Recognize High, with Digital Filter  | 3 * N * TCY         | —    | ns    | N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 3</b> ) |

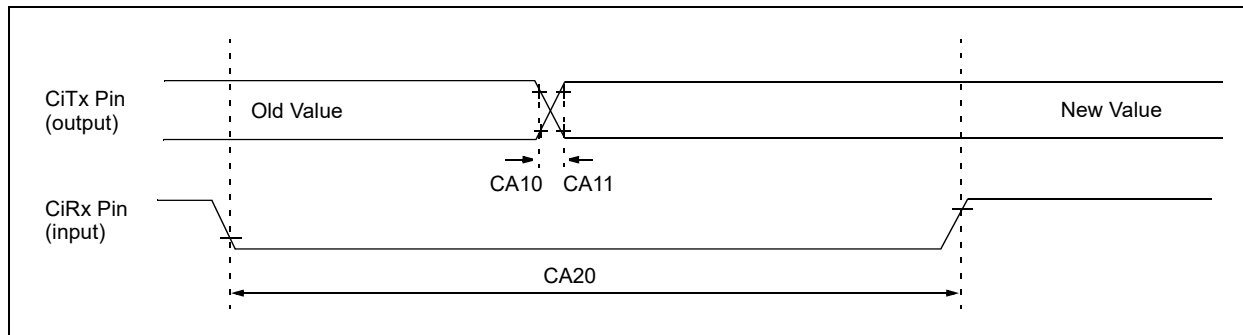
**Note 1:** These parameters are characterized but not tested in manufacturing.

**2:** Data in “Typical” column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**3:** N = Index Channel Digital Filter Clock Divide Select bits.

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-20: CANFDx MODULE I/O TIMING CHARACTERISTICS**



**TABLE 36-38: CANFDx MODULE I/O TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                     |      |       |                                    |
|--------------------|--------|---|---|---------------------|------|-------|------------------------------------|
| Param No.          | Symbol | Characteristic <sup>(1)</sup>             | Min.  | Typ. <sup>(2)</sup> | Max. | Units | Conditions                         |
| CA10               | TioF   | Port Output Fall Time                     | —   | —                   | —    | ns    | See parameter <a href="#">DO32</a> |
| CA11               | TioR   | Port Output Rise Time                     | —   | —                   | —    | ns    | See parameter <a href="#">DO31</a> |
| CA20               | Tcwf   | Pulse Width to Trigger CAN Wake-up Filter | 700   | —                   | —    | ns    | —                                  |

- Note 1:** These parameters are characterized but not tested in manufacturing.
- 2:** Data in “Typical” column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
- 3:** CAN module wake from sleep could take up to 300us worst case. Multiple CAN messages could therefore be missed depending on the USER CAN baud rate.
- 4:** Recommended CAN-FD clock frequency is 20MHz/40MHz/80MHz. Implementing following steps to set Module Clock frequency of 40MHz while maintaining CPU clock frequency of 120MHz:
- Select module clock source as REFCLK4 by configuring CLKSEL0 bit (CFDxCON<7>) = 1
  - Select SYSCLK as the input clock fro REFCLK4 by configure ROSEL bits (REFO4CON<3:0>) = 0
  - Set REF4CLK prescaler to 3 by configuring RODIV bits (REFO4CON<30:16>) = 1 and ROTRIM bits (REFO4TRIM<31:23> = 256.



# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-39: ADC MODULE SPECIFICATIONS**

| AC CHARACTERISTICS   |           |  | Standard Operating Conditions: 2.5V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |      |                            |       |  |
|--|-----------|--|---|------|----------------------------|-------|--|
| Param. No.   | Symbol    | Characteristics                            | Min.  | Typ. | Max.                       | Units | Conditions   |
| <b>Device Supply</b>   |           |  |   |      |                            |       |  |
| AD01   | AVDD      | Module VDD Supply                          | Greater of VDD - 0.3 or 2.3   | —    | Lesser of VDD + 0.3 or 3.6 | V     | —  |
| AD02   | AVSS      | Module VSS Supply                          | VSS   | —    | VSS + 0.3                  | V     | —  |
| <b>Reference Inputs</b>                                      |           |  |   |      |                            |       |  |
| AD05   | VREFH     | Reference Voltage High                     | VREFL + 1.8   | —    | AVDD                       | V     | (Note 1)   |
| AD06   | VREFL     | Reference Voltage Low                      | AVSS  | —    | VREFH - 1.8                | V     | (Note 1)   |
| AD07   | VREF      | Absolute Reference Voltage (VREFH - VREFL) | 1.8   | —    | AVDD                       | V     | (Note 2)   |
| AD08   | IREF      | Current Drain                              | —   | 102  | —                          | μA    | ADC is operating or is in Stand-by.  |
|  | IVREF     | Bandgap Reference                          | 1.145   | 1.2  | 1.255                      | V     |  |
| <b>Analog Input</b>  |           |  |   |      |                            |       |  |
| AD12   | VINH-VINL | Full-Scale Input Span                      | VREFL   | —    | VREFH                      | V     | —  |
| AD13   | VINL      | Absolute VINL Input Voltage                | AVSS  | —    | VREFL                      | V     | —  |
| AD14   | VINH      | Absolute VINH Input Voltage                | AVSS  | —    | VREFH                      | V     | —  |
| <b>ADC Accuracy – Measurements with External VREF+/VREF-</b> |           |  |   |      |                            |       |  |
| AD20c  | Nr        | Resolution                                 | 6   | —    | 12                         | bits  | Selectable 6, 8, 10, 12 Resolution Ranges  |
| AD21c  | INL       | Integral Nonlinearity                      | -3.5  | ±1.5 | +3.5                       | LSb   | AVSS = VREFL = 0V, AVDD = VREFH = 3.3V (Note 4)                                    |
| AD22c  | DNL       | Differential Nonlinearity                  | -1  | 1.2  | 3                          | LSb   | AVSS = VREFL = 0V, AVDD = VREFH = 3.3V (Note 4)                                    |
| AD23c  | GERR      | Gain Error                                 | -8  | 1    | 8                          | LSb   | AVSS = VREFL = 0V, AVDD = VREFH = 3.3V, Gain error includes offset error. (Note 4) |
| AD24c  | EOFF      | Offset Error                               | -5  | -0.5 | 5                          | LSb   | AVSS = 0V, AVDD = 3.3V (Note 4)  |
| AD25c  | —         | Monotonicity                               | —   | —    | —                          | —     | Guaranteed (Note 2)  |
| <b>Dynamic Performance</b>                                   |           |  |   |      |                            |       |  |
| AD31b  | SINAD     | Signal to Noise and Distortion             | —   | 67   | —                          | dB    | Single-ended (Notes 2,3)   |
| AD34b  | ENOB      | Effective Number of bits                   | —   | 10.8 | —                          | bits  | (Notes 2,3)  |

**Note 1:** These parameters are not characterized or tested in manufacturing.

**2:** These parameters are characterized, but not tested in manufacturing.

**3:** Characterized with a 1 kHz sine wave.

**4:** Tested on single ADC core, operating in Single-ended mode.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-40: ANALOG-TO-DIGITAL CONVERSION TIMING REQUIREMENTS**

| AC CHARACTERISTICS <sup>(2)</sup> |        |   | Standard Operating Conditions: 2.5V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |                     |                               |                          |  |
|-----------------------------------|--------|---|---|---------------------|-------------------------------|--------------------------|--|
| Param. No.                        | Symbol | Characteristics                                   | Min.  | Typ. <sup>(1)</sup> | Max.                          | Units                    | Conditions   |
| <b>Clock Parameters</b>           |        |   |   |                     |                               |                          |  |
| AD50                              | TAD    | ADC Clock Period                                  | 16.667  | —                   | 6250                          | ns                       | —  |
| <b>Throughput Rate</b>            |        |   |   |                     |                               |                          |  |
| AD51                              | FTP    | Sample Rate for ADC0-ADC5 (Class 1 Inputs)        | —   | —                   | 3.75<br>4.284<br>4.992<br>6   | Msp<br>Msp<br>Msp<br>Msp | 12-bit resolution Source Impedance $\leq 200\Omega$<br>10-bit resolution Source Impedance $\leq 200\Omega$<br>8-bit resolution Source Impedance $\leq 200\Omega$<br>6-bit resolution Source Impedance $\leq 200\Omega$         |
|                                   |        | Sample Rate for ADC7 (Class 2 and Class 3 Inputs) | —   | —                   | 3.53<br>4.00<br>4.615<br>5.45 | Msp<br>Msp<br>Msp<br>Msp | 12-bit resolution Source Impedance $\leq 200\Omega$<br>10-bit resolution Source Impedance $\leq 200\Omega$<br>8-bit resolution Source Impedance $\leq 200\Omega$<br>6-bit resolution Source Impedance $\leq 200\Omega$         |
| <b>Timing Parameters</b>          |        |   |   |                     |                               |                          |  |
| AD60                              | TSAMP  | Sample Time for ADC0-ADC5 (Class 1 Inputs)        | 3<br>4<br>5<br>13   | —                   | —                             | TAD                      | Source Impedance $\leq 200\Omega$ , Max ADC clock<br>Source Impedance $\leq 500\Omega$ , Max ADC clock<br>Source Impedance $\leq 1\text{ K}\Omega$ , Max ADC clock<br>Source Impedance $\leq 5\text{ K}\Omega$ , Max ADC clock |
|                                   |        | Sample Time for ADC7 (Class 2 and Class 3 Inputs) | 4<br>5<br>6<br>14   | —                   | —                             | TAD                      | Source Impedance $\leq 200\Omega$ , Max ADC clock<br>Source Impedance $\leq 500\Omega$ , Max ADC clock<br>Source Impedance $\leq 1\text{ K}\Omega$ , Max ADC clock<br>Source Impedance $\leq 5\text{ K}\Omega$ , Max ADC clock |
|                                   |        | Sample Time for ADC7 (Class 2 and Class 3 Inputs) | See <a href="#">Table 36-41</a>   | —                   | —                             | TAD                      | CVDEN (ADCCON1<11>) = 1  |
| AD62                              | TCONV  | Conversion Time (after sample time is complete)   | —<br>—<br>—<br>—  | —<br>—<br>—<br>—    | 13<br>11<br>9<br>7            | TAD                      | 12-bit resolution<br>10-bit resolution<br>8-bit resolution<br>6-bit resolution   |
| AD65                              | TWAKE  | Wake-up time from Low-Power Mode                  | —   | 500                 | —                             | TAD                      | Lesser of 500 TAD or 20 $\mu\text{s}$  |
|                                   |        |   | —   | 20                  | —                             | $\mu\text{s}$            |  |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**Note 2:** The ADC module is functional at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$ , but with degraded performance. Unless otherwise stated, module functionality is guaranteed, but not characterized.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-41: ADC SAMPLE TIMES WITH CVD ENABLED**

| AC CHARACTERISTICS <sup>(2)</sup> |        |  | Standard Operating Conditions: 2.5V to 3.6V<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |                                    |      |                                    |                                    |
|-----------------------------------|--------|--|---|------------------------------------|------|------------------------------------|------------------------------------|
| Param. No.                        | Symbol | Characteristics  | Min.  | Typ. <sup>(1)</sup>                | Max. | Units                              | Conditions                         |
| AD60a                             | TSAMP  | Sample Time for ADC7 (Class 2 and Class 3 Inputs) with the CVDEN bit (ADCCON1<11>) = 1 | 8   |                                    |      | TAD                                | Source Impedance ≤ 200Ω            |
|                                   |        |  | 9   |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 001 |
|                                   |        |  | 11  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 010 |
|                                   |        |  | 12  | —                                  | —    |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 011 |
|                                   |        |  | 14  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 100 |
|                                   |        |  | 16  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 101 |
|                                   |        |  | 17  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 110 |
|                                   |        |  | 17  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 111 |
|                                   |        |  | 10  |                                    |      | TAD                                | Source Impedance ≤ 500Ω            |
|                                   |        |  | 12  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 001 |
|                                   |        |  | 14  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 010 |
|                                   |        |  | 16  | —                                  | —    |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 011 |
|                                   |        |  | 18  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 100 |
|                                   |        |  | 19  |                                    |      |                                    | CVDCPL<2:0> (ADCCON2<28:26>) = 101 |
|                                   |        |  | 21  |                                    |      | CVDCPL<2:0> (ADCCON2<28:26>) = 110 |                                    |
|                                   |        |  | 21  |                                    |      | CVDCPL<2:0> (ADCCON2<28:26>) = 111 |                                    |
| 13                                |        |  | TAD   | Source Impedance ≤ 1 KΩ            |      |                                    |                                    |
| 16                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 001 |      |                                    |                                    |
| 18                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 010 |      |                                    |                                    |
| 21                                | —      | —  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 011 |      |                                    |                                    |
| 23                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 100 |      |                                    |                                    |
| 26                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 101 |      |                                    |                                    |
| 28                                |        |  | CVDCPL<2:0> (ADCCON2<28:26>) = 110  |                                    |      |                                    |                                    |
| 28                                |        |  | CVDCPL<2:0> (ADCCON2<28:26>) = 111  |                                    |      |                                    |                                    |
| 41                                |        |  | TAD   | Source Impedance ≤ 5 KΩ            |      |                                    |                                    |
| 48                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 001 |      |                                    |                                    |
| 56                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 010 |      |                                    |                                    |
| 63                                | —      | —  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 011 |      |                                    |                                    |
| 70                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 100 |      |                                    |                                    |
| 78                                |        |  |   | CVDCPL<2:0> (ADCCON2<28:26>) = 101 |      |                                    |                                    |
| 85                                |        |  | CVDCPL<2:0> (ADCCON2<28:26>) = 110  |                                    |      |                                    |                                    |
| 85                                |        |  | CVDCPL<2:0> (ADCCON2<28:26>) = 111  |                                    |      |                                    |                                    |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**Note 2:** The ADC module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is guaranteed, but not characterized.

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-42: CONTROL DAC (CDAC) SPECIFICATIONS**

| AC CHARACTERISTICS |          |   | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |        |                   |               |  |
|--------------------|----------|---|---|--------|-------------------|---------------|--|
| Param. No.         | Symbol   | Characteristics   | Min.  | Typ.   | Max.              | Units         | Conditions   |
| <b>CDAC</b>        |          |   |   |        |                   |               |  |
| CD10               | VOUT     | CDAC Output Voltage Range for Guaranteed Settling Time Specifications | 0.1 *<br>CDACVREF   | —      | 0.9 *<br>CDACVREF | V             | @ ILOAD = IOUT (max)   |
| CD11               | N        | CDAC Resolution   | 12  | —      | —                 | Bits          | Guaranteed Monotonic by architecture   |
| CD12               | INL      | CDAC Integral Nonlinearity  | -4  | 1.03   | +4                | LSB           | Guaranteed Monotonic by architecture with CDACVREF = AVDD = 3.3V   |
| CD13               | DNL      | CDAC Differential Nonlinearity  | -1  | -0.2   | +2                | LSB           | Guaranteed Monotonic by architecture with CDACVREF = AVDD = 3.3V   |
| CD14               | OERR     | CDAC Offset Error   | -13   | —      | 13                | mV            | CDACVREF = AVDD = 3.3V   |
| CD15               | GERR     | CDAC Gain Error   | -1  | -0.088 | +1                | % of FS       | CDACVREF = AVDD = 3.3V   |
| CD16               | CDACVREF | CDAC VREF Input Range   | 0.5   | —      | AVDD              | V             | —  |
| CD17               | TON      | CDAC Module Turn On Time  | —   | 1.0    | 2                 | $\mu\text{s}$ | From write of DACON bit  |
| CD18               | TOFF     | CDAC Module Turn Off Time   | —   | 1.0    | 2                 | $\mu\text{s}$ | From write of DACON bit  |
| CD19               | TST      | Settling Time   | —   | 3      | 6                 | $\mu\text{s}$ | Output is within $\pm 4$ LSb of desired output step voltage with a 10% to 90% step or 90% to 10% step. With load capacitance of 30 pF. PBCLK2 > 15 MHz |
| CD20               | Fs       | Sampling Frequency  | —   | —      | 1                 | Msp           | Maximum frequency for a correct CDAC output change for small variations of input codes (from code to code plus 1 LSb).                                 |
| CD21               | CLOAD    | Output Load Capacitance   | ---   | —      | 30                | pF            | User application loads   |
| DC22               | IOUT     | Output Current Drive Strength   | —   | —      | 1                 | mA            | Sink and source  |

# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-43: CTMU CURRENT SOURCE SPECIFICATIONS**

| AC CHARACTERISTICS         |        |  | Standard Operating Conditions: 2.3V to 3.6V (Note 1)<br>(unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for Industrial<br>-40°C ≤ TA ≤ +125°C for Extended |       |      |       |                                   |
|----------------------------|--------|--|--|-------|------|-------|-----------------------------------|
| Param No.                  | Symbol | Characteristic                                     | Min.   | Typ.  | Max. | Units | Conditions                        |
| <b>CTMU CURRENT SOURCE</b> |        |  |  |       |      |       |                                   |
| CTMU0                      | RES    | Resolution   | -2   | —     | +2   | °C    | 3.3V @ -40°C to 125°C             |
| CTMUI1                     | IOUT1  | Base Range <sup>(1)</sup>                          | —  | 0.55  | —    | μA    | CTMUCON <1:0> = 01                |
| CTMUI2                     | IOUT2  | 10x Range <sup>(1)</sup>                           | —  | 5.5   | —    | μA    | CTMUCON <1:0> = 10                |
| CTMUI3                     | IOUT3  | 100x Range <sup>(1)</sup>                          | —  | 55    | —    | μA    | CTMUCON <1:0> = 11                |
| CTMUI4                     | IOUT4  | 1000x Range <sup>(1)</sup>                         | —  | 550   | —    | μA    | CTMUCON <1:0> = 00                |
| CTMUFV1                    | VF     | Temperature Diode Forward Voltage <sup>(1,2)</sup> | —  | 0.598 | —    | V     | TA = +25°C,<br>CTMUCON <1:0> = 01 |
|                            |        |  | —  | 0.658 | —    | V     | TA = +25°C,<br>CTMUCON <1:0> = 10 |
|                            |        |  | —  | 0.721 | —    | V     | TA = +25°C,<br>CTMUCON <1:0> = 11 |
| CTMUFV2                    | VFVR   | Temperature Diode Rate of Change <sup>(1,2)</sup>  | —  | -1.74 | —    | mV/°C | CTMUCON <1:0> = 01                |
|                            |        |  | —  | -1.58 | —    | mV/°C | CTMUCON <1:0> = 10                |
|                            |        |  | —  | -1.42 | —    | mV/°C | CTMUCON <1:0> = 11                |

**Note 1:** Nominal value at center point of current trim range (CTMUCON<15:10> = 000000).

- 2:** Parameters are characterized but not tested in manufacturing. Measurements taken with the following conditions:
- VREF+ = AVDD = 3.3V
  - ADC module configured for conversion speed of 500 ksps
  - All PMD bits are cleared (PMDx = 0)
  - Executing a `while(1)` statement
  - Device operating from the FRC with no PLL

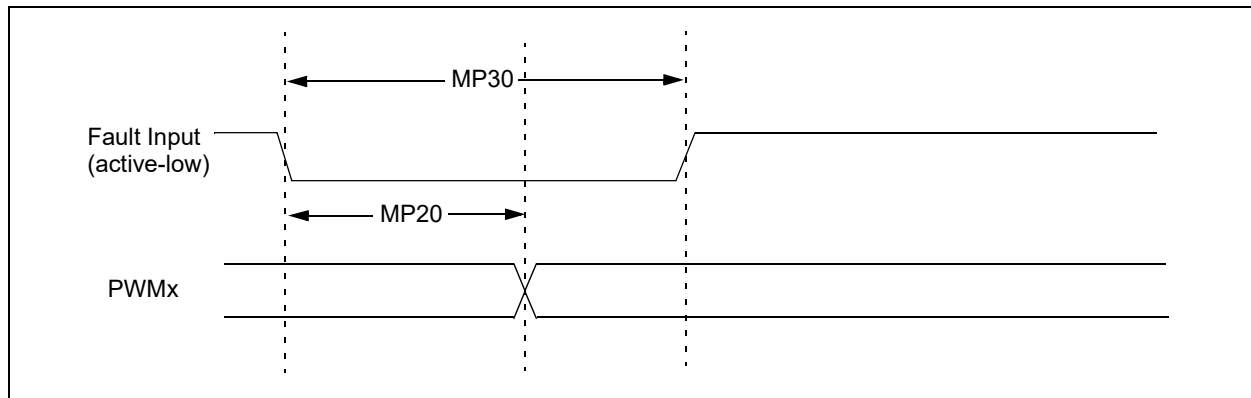
# PIC32MK GPG/MCJ with CAN FD Family

**TABLE 36-44: UART TIMING CHARACTERISTICS**

| AC CHARACTERISTICS |        |                                | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial |      |      |       |  |
|--------------------|--------|--------------------------------|--|------|------|-------|--|
| Param. No.         | Symbol | Characteristics <sup>(1)</sup> | Min.   | Typ. | Max. | Units | Conditions   |
| UT10               | FB     | Baud Rate<br>BRGH = 0          | —  | —    | 7.5  | Mbps  | Baud rate = (FPBy / (16 * (UxBRG + 1)))<br>where:<br>'x' = 1-6<br>'y' = FPBCLK2 for UART1 and UART2<br>'y' = FPBLKC3 for UART3-UART6 |
| UT20               |        | BRGH = 1                       | 7.5  | —    | 30   | Mbps  | Baud rate = (FPBy / (4 * (UxBRG + 1)))<br>where:<br>'x' = 1-6<br>'y' = FPBCLK2 for UART1 and UART2<br>'y' = FPBLKC3 for UART3-UART6  |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

**FIGURE 36-21: MOTOR CONTROL PWM MODULE FAULT TIMING CHARACTERISTICS**



**TABLE 36-45: MOTOR CONTROL PWM MODULE TIMING REQUIREMENTS**

| AC CHARACTERISTICS |        |                                 | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for Commercial<br>-40°C ≤ TA ≤ +85°C for Industrial |      |      |       |                                    |
|--------------------|--------|---------------------------------|--|------|------|-------|------------------------------------|
| Param No.          | Symbol | Characteristic <sup>(1)</sup>   | Min.   | Typ. | Max. | Units | Conditions                         |
| MP10               | TFPWM  | PWM Output Fall Time            | —  | —    | —    | ns    | See parameter <a href="#">DO32</a> |
| MP11               | TRPWM  | PWM Output Rise Time            | —  | —    | —    | ns    | See parameter <a href="#">DO31</a> |
| MP20               | TFD    | Fault Input ↓ to PWM I/O Change | —  | —    | 50   | ns    | —                                  |
| MP30               | TFH    | Fault Input Pulse Width         | 50   | —    | —    | ns    | —                                  |

**Note 1:** These parameters are characterized, but not tested in manufacturing.

# PIC32MK GPG/MCJ with CAN FD Family

FIGURE 36-22: LOW VOLTAGE DETECT CHARACTERISTICS

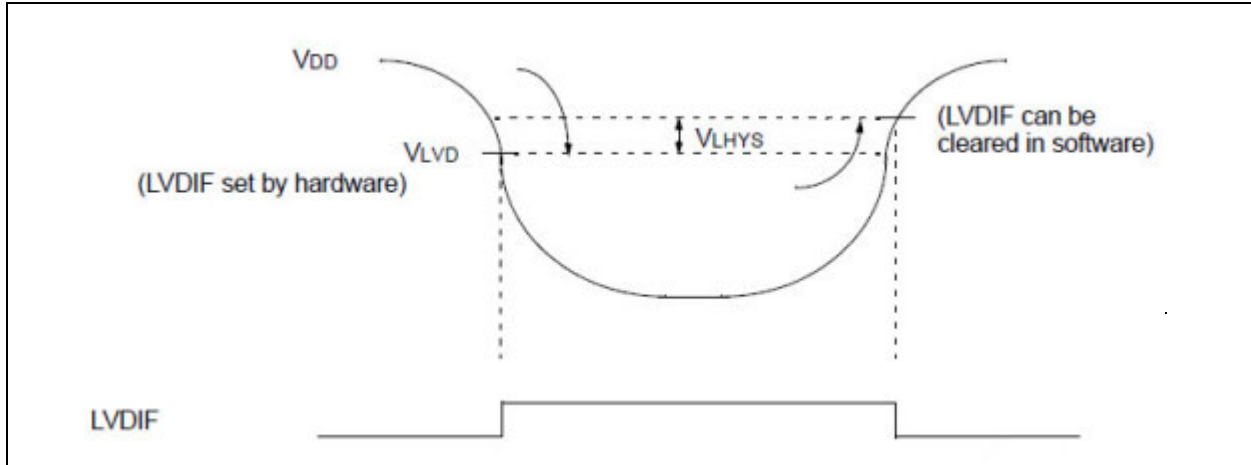
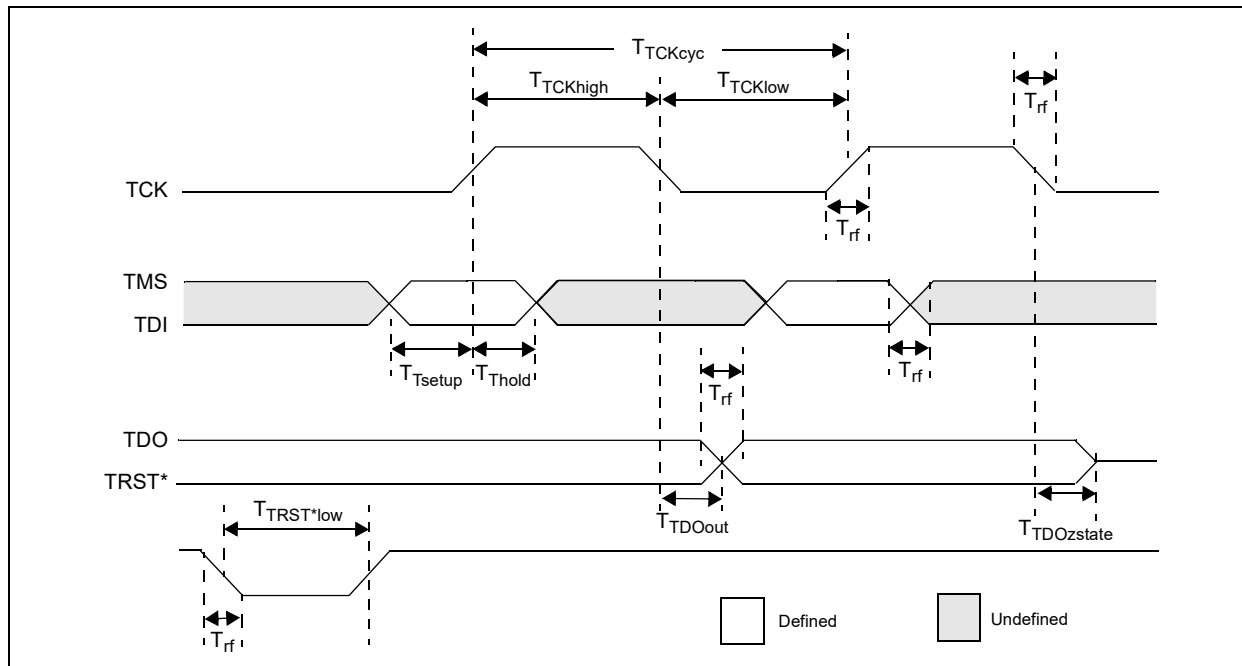


TABLE 36-46: ELECTRICAL CHARACTERISTICS LVD

| DC Specifications |   | Standard Operating Conditions: 2.3V to 3.6V<br>(Unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$ |                  |          |       |  |          |
|-------------------|---|--|------------------|----------|-------|--|----------|
| Symbol            | Characteristic                            | Min  | Typ <sup>1</sup> | Max      | Units | Conditions                                     |          |
| VLVD              | LVD Voltage on VDD transition high to low | LVDL = 0000  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 0001  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 0010  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 0011  | —                | —        | —     | V  | Reserved |
|                   |   | LVD = 0100   | 3.38             | 3.52     | 3.66  | V  |          |
|                   |   | LVD = 0101   | 3.14             | 3.27     | 3.4   | V  |          |
|                   |   | LVD = 0110   | 2.87             | 2.99     | 3.11  | V  |          |
|                   |   | LVD = 0111   | 2.67             | 2.78     | 2.89  | V  |          |
|                   |   | LVD = 1000   | 2.57             | 2.68     | 2.79  | V  |          |
|                   |   | LVD = 1001   | 2.39             | 2.49     | 2.59  | V  |          |
|                   |   | LVD = 1010   | 2.29             | 2.39     | 2.48  | V  |          |
|                   |   | LVDL = 1011  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 1100  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 1101  | —                | —        | —     | V  | Reserved |
|                   |   | LVDL = 1110  | —                | —        | —     | V  | Reserved |
|                   | LVDL = 1111                               | External   | External         | External | V     |  |          |
| VLVERR            | LVD Total Error                           | -10  | -                | 10       | mV    | offset, regulation drift and temperature drift |          |
| VLHYS             | Low-Voltage Detect Hysteresis             | -  | 6.8              | 24       | mV    |  |          |

# PIC32MK GPG/MCJ with CAN FD Family

**FIGURE 36-23: EJTAG TIMING CHARACTERISTICS**



**TABLE 36-47: EJTAG TIMING REQUIREMENTS**

| AC CHARACTERISTICS |            |  | Standard Operating Conditions: 2.3V to 3.6V<br>(unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended |      |       |            |
|--------------------|------------|--|---|------|-------|------------|
| Param. No.         | Symbol     | Description <sup>(1)</sup>                       | Min.  | Max. | Units | Conditions |
| EJ1                | TTCKCYC    | TCK Cycle Time                                   | 25  | —    | ns    | —          |
| EJ2                | TTCKHIGH   | TCK High Time                                    | 10  | —    | ns    | —          |
| EJ3                | TTCKLOW    | TCK Low Time                                     | 10  | —    | ns    | —          |
| EJ4                | TTSETUP    | TAP Signals Setup Time Before Rising TCK         | 5   | —    | ns    | —          |
| EJ5                | TTHOLD     | TAP Signals Hold Time After Rising TCK           | 3   | —    | ns    | —          |
| EJ6                | TTDOOUT    | TDO Output Delay Time from Falling TCK           | —   | 5    | ns    | —          |
| EJ7                | TTDOZSTATE | TDO 3-State Delay Time from Falling TCK          | —   | 5    | ns    | —          |
| EJ8                | TTRSTLOW   | TRST Low Time                                    | 25  | —    | ns    | —          |
| EJ9                | TRF        | TAP Signals Rise/Fall Time, All Input and Output | —   | —    | ns    | —          |

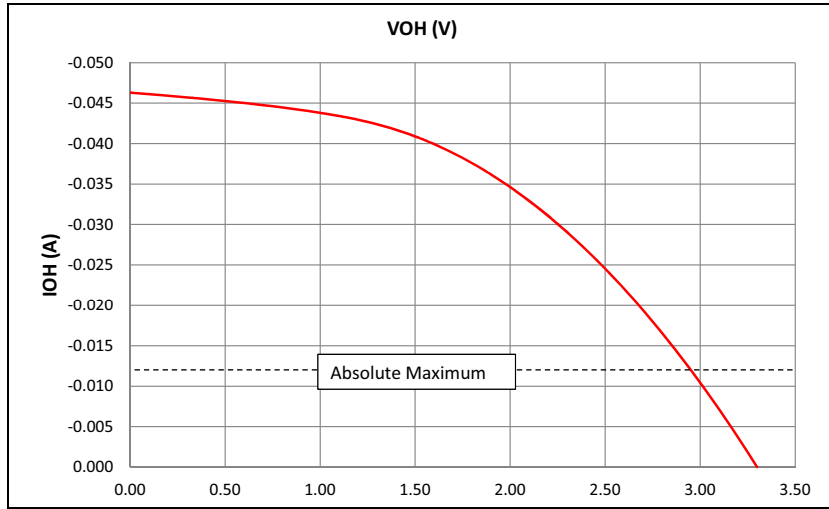
**Note 1:** These parameters are characterized, but not tested in manufacturing.



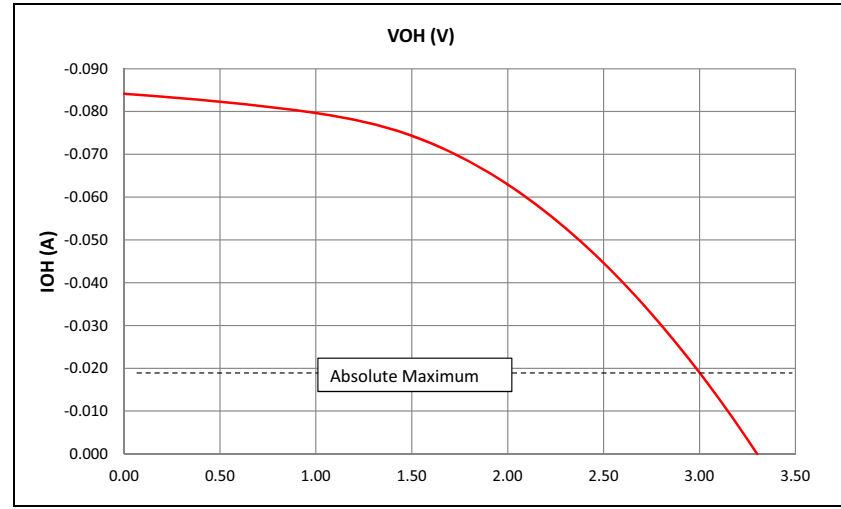
### 37.0 AC AND DC CHARACTERISTICS GRAPHS

**Note:** The graphs provided are a statistical summary based on a limited number of samples and are provided for design guidance purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

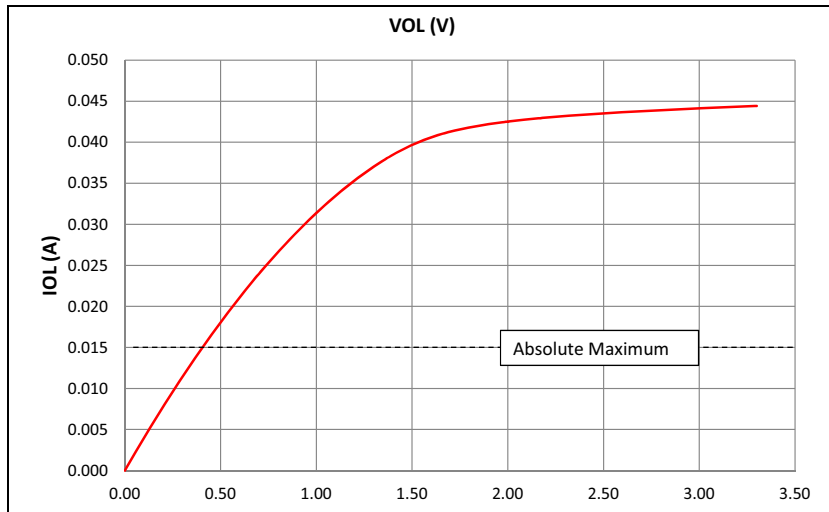
**FIGURE 37-1:  $V_{OH}$  – 4x DRIVER PINS**



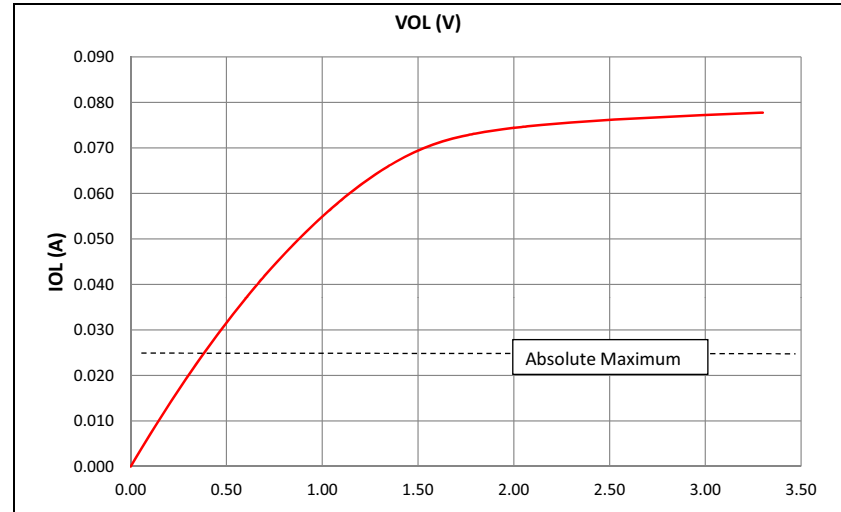
**FIGURE 37-3:  $V_{OH}$  – 8x DRIVER PINS**



**FIGURE 37-2:  $V_{OL}$  – 4x DRIVER PINS**



**FIGURE 37-4:  $V_{OL}$  – 8x DRIVER PINS**



# PIC32MK GPG/MCJ with CAN FD Family

---

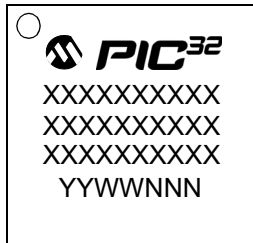
NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

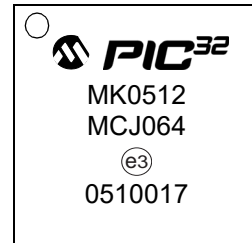
## 38.0 PACKAGING INFORMATION

### 38.1 Package Marking Information

64-Lead QFN (9x9x0.9 mm)



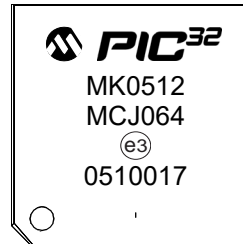
Example



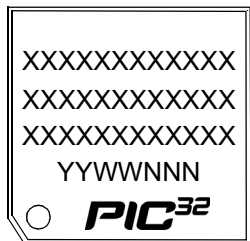
64-Lead TQFP (10x10x1 mm)



Example



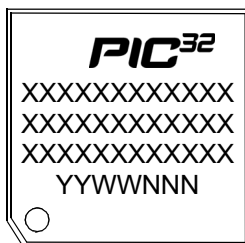
48-Lead TQFP (7x7x1 mm)



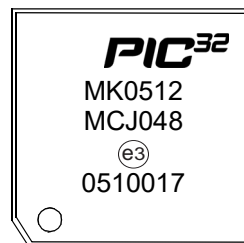
Example



48-Lead VQFN (6x6x0.9 mm)



Example



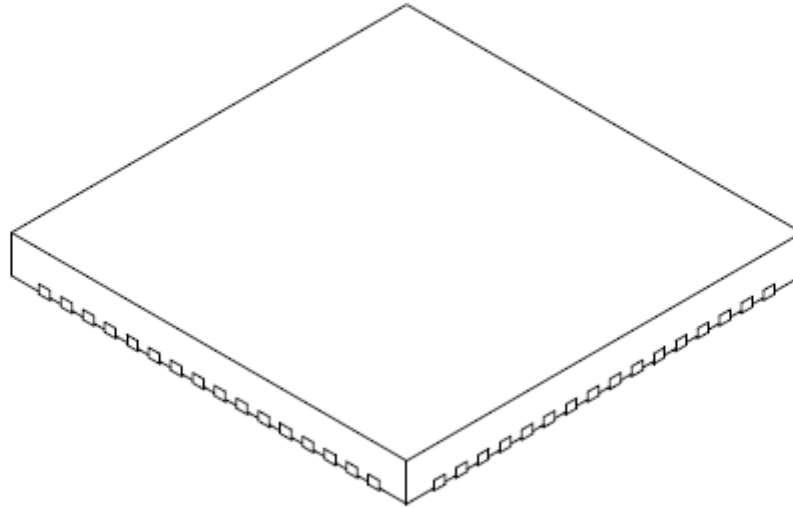
|                |   |  |
|----------------|---|--|
| <b>Legend:</b> | XX...X  | Customer-specific information  |
|                | Y   | Year code (last digit of calendar year)  |
|                | YY  | Year code (last 2 digits of calendar year)   |
|                | WW  | Week code (week of January 1 is week '01')   |
|                | NNN   | Alphanumeric traceability code   |
|                | *   | Pb-free JEDEC designator for Matte Tin (Sn)  |
|                |   | This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package. |
| <b>Note:</b>   | In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information. |  |



# PIC32MK GPG/MCJ with CAN FD Family

**64-Lead Plastic Quad Flat, No Lead Package (R4X) – 9x9x0.9 mm Body with 5.40 x 5.40 Exposed Pad [QFN]**

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension Limits       | Units | MILLIMETERS |      |      |
|------------------------|-------|-------------|------|------|
|                        |       | MIN         | NOM  | MAX  |
| Number of Pins         | N     | 64          |      |      |
| Pitch                  | e     | 0,50 BSC    |      |      |
| Overall Height         | A     | 0.80        | 0.90 | 1.00 |
| Standoff               | A1    | 0,00        | 0,02 | 0,05 |
| Contact Thickness      | A3    | 0.20 REF    |      |      |
| Overall Width          | E     | 9,00 BSC    |      |      |
| Exposed Pad Width      | E2    | 5.30        | 5.40 | 5.50 |
| Overall Length         | D     | 9,00 BSC    |      |      |
| Exposed Pad Length     | D2    | 5.30        | 5.40 | 5.50 |
| Contact Width          | b     | 0.20        | 0.25 | 0.30 |
| Contact Length         | L     | 0.30        | 0.40 | 0.50 |
| Contact-to-Exposed Pad | K     | 0.20        | -    | -    |

**Notes:**

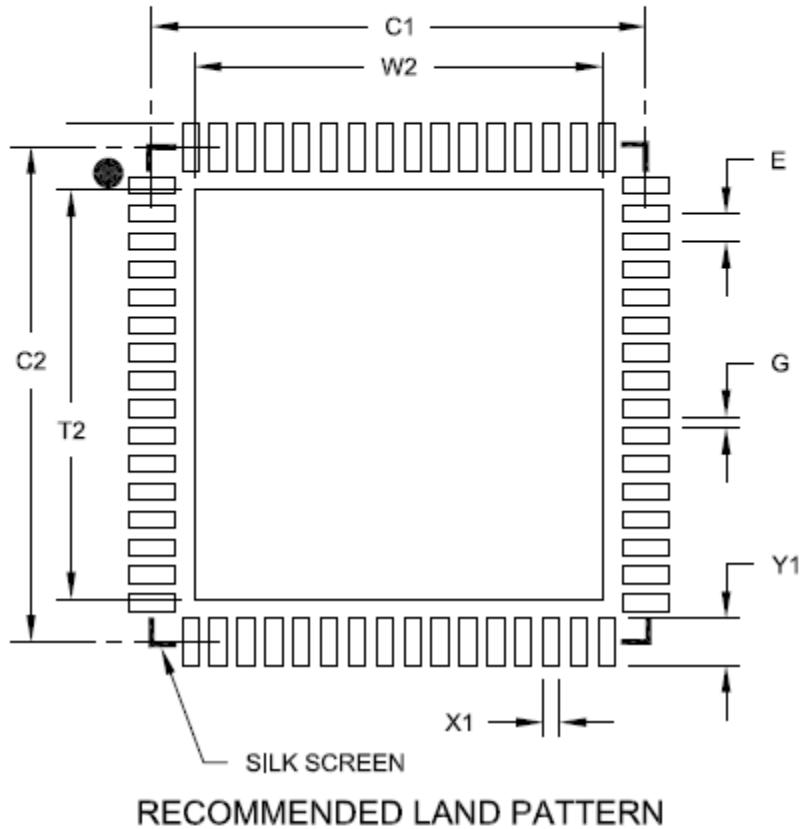
1. Pin 1 visual Index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14,5M,  
 BSC; Basic Dimension, Theoretically exact value shown without tolerances,  
 REF; Reference Dimension, usually without tolerance, for Information purposes only.

Microchip Technology Drawing C04-154A Sheet 2 of 2

# PIC32MK GPG/MCJ with CAN FD Family

64-Lead Plastic Quad Flat, No Lead Package (R4X) – 9x9x0.9 mm Body [QFN]  
With 0.40 mm Contact Length

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packageing>



| Dimension Limits           | Units | MILLIMETERS |      |      |
|----------------------------|-------|-------------|------|------|
|                            |       | MIN         | NOM  | MAX  |
| Contact Pitch              | E     | 0.50 BSC    |      |      |
| Optional Center Pad Width  | W2    |             |      | 7.35 |
| Optional Center Pad Length | T2    |             |      | 7.35 |
| Contact Pad Spacing        | C1    |             | 8.90 |      |
| Contact Pad Spacing        | C2    |             | 8.90 |      |
| Contact Pad Width (X64)    | X1    |             |      | 0.30 |
| Contact Pad Length (Y64)   | Y1    |             |      | 0.85 |
| Distance Between Pads      | G     | 0.20        |      |      |

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

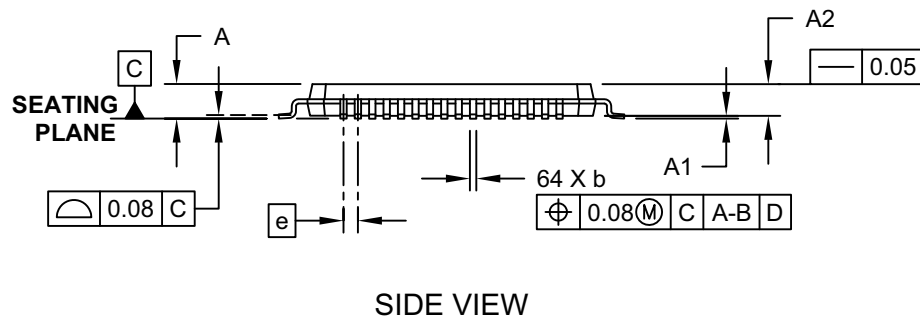
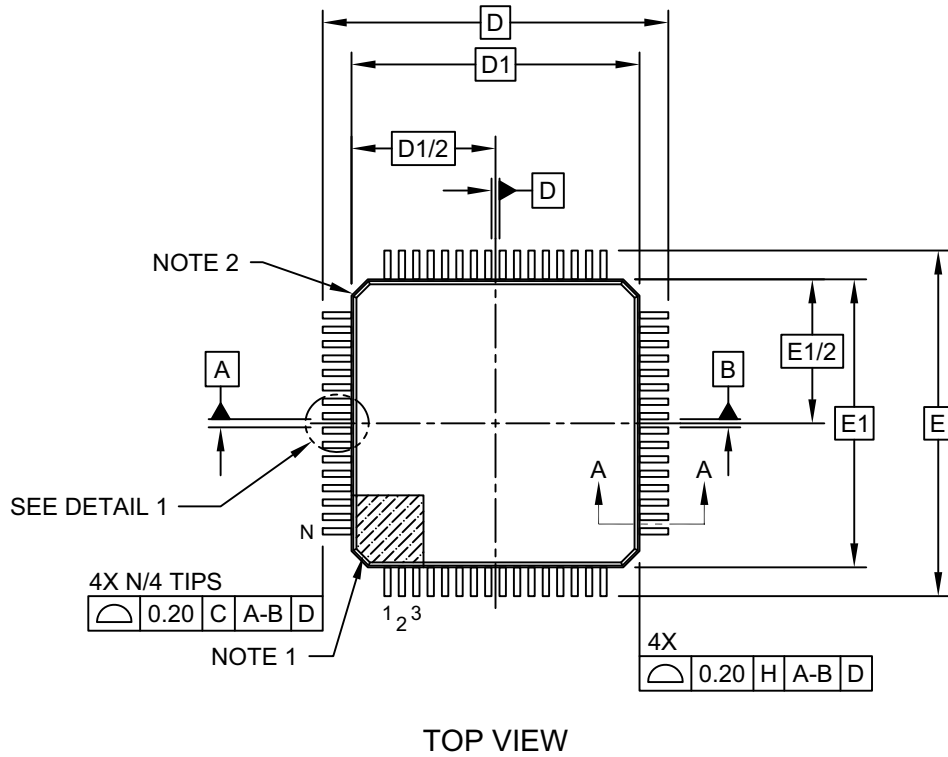
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2149A

# PIC32MK GPG/MCJ with CAN FD Family

## 64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

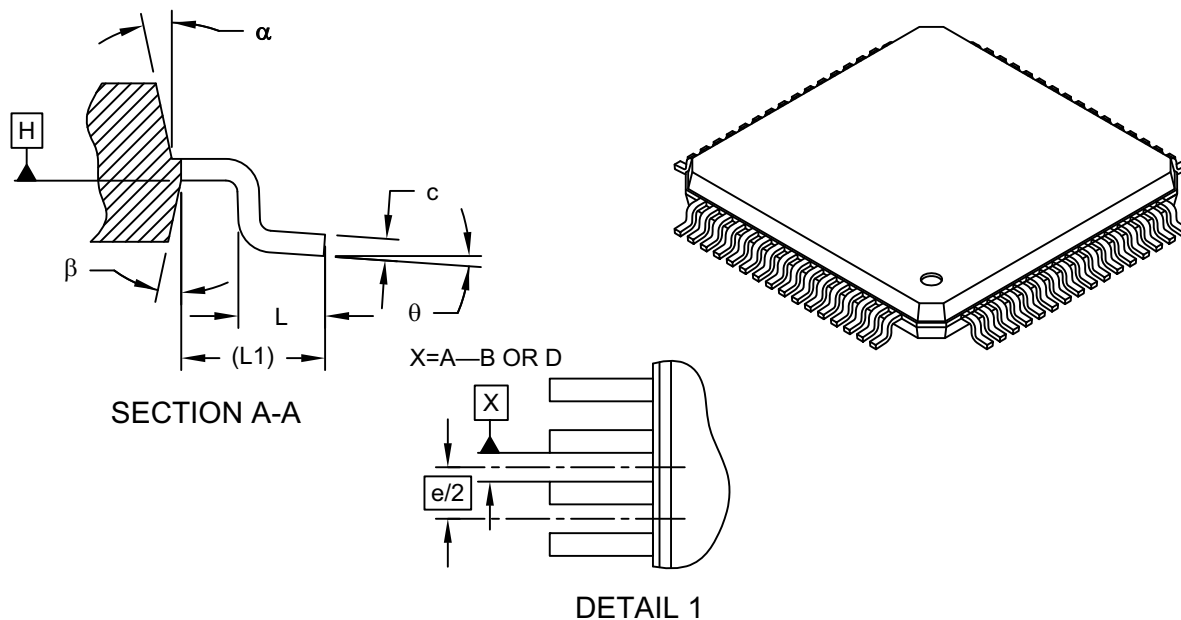


Microchip Technology Drawing C04-085C Sheet 1 of 2

# PIC32MK GPG/MCJ with CAN FD Family

## 64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension Limits         | Units    | MILLIMETERS |      |      |
|--------------------------|----------|-------------|------|------|
|                          |          | MIN         | NOM  | MAX  |
| Number of Leads          | N        | 64          |      |      |
| Lead Pitch               | e        | 0.50 BSC    |      |      |
| Overall Height           | A        | -           | -    | 1.20 |
| Molded Package Thickness | A2       | 0.95        | 1.00 | 1.05 |
| Standoff                 | A1       | 0.05        | -    | 0.15 |
| Foot Length              | L        | 0.45        | 0.60 | 0.75 |
| Footprint                | L1       | 1.00 REF    |      |      |
| Foot Angle               | $\phi$   | 0°          | 3.5° | 7°   |
| Overall Width            | E        | 12.00 BSC   |      |      |
| Overall Length           | D        | 12.00 BSC   |      |      |
| Molded Package Width     | E1       | 10.00 BSC   |      |      |
| Molded Package Length    | D1       | 10.00 BSC   |      |      |
| Lead Thickness           | c        | 0.09        | -    | 0.20 |
| Lead Width               | b        | 0.17        | 0.22 | 0.27 |
| Mold Draft Angle Top     | $\alpha$ | 11°         | 12°  | 13°  |
| Mold Draft Angle Bottom  | $\beta$  | 11°         | 12°  | 13°  |

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
- Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

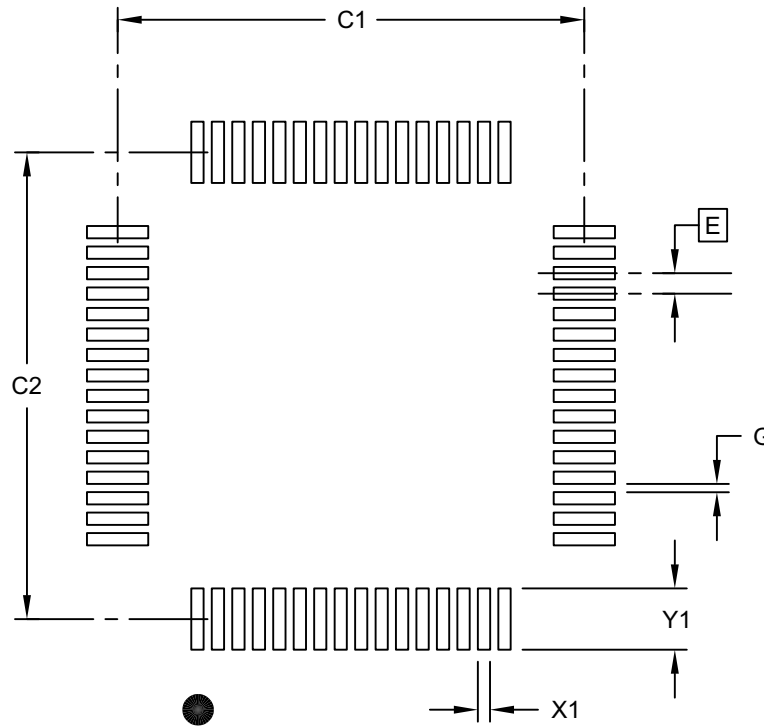
Microchip Technology Drawing C04-085C Sheet 2 of 2



# PIC32MK GPG/MCJ with CAN FD Family

## 64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

| Dimension Limits         | Units | MILLIMETERS |       |      |
|--------------------------|-------|-------------|-------|------|
|                          |       | MIN         | NOM   | MAX  |
| Contact Pitch            | E     | 0.50 BSC    |       |      |
| Contact Pad Spacing      | C1    |             | 11.40 |      |
| Contact Pad Spacing      | C2    |             | 11.40 |      |
| Contact Pad Width (X28)  | X1    |             |       | 0.30 |
| Contact Pad Length (X28) | Y1    |             |       | 1.50 |
| Distance Between Pads    | G     | 0.20        |       |      |

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

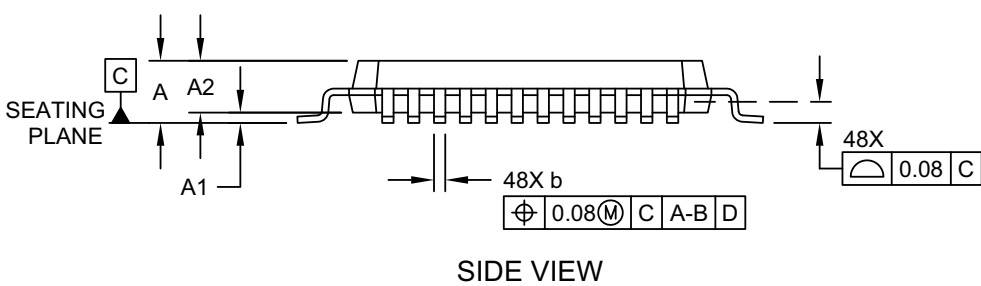
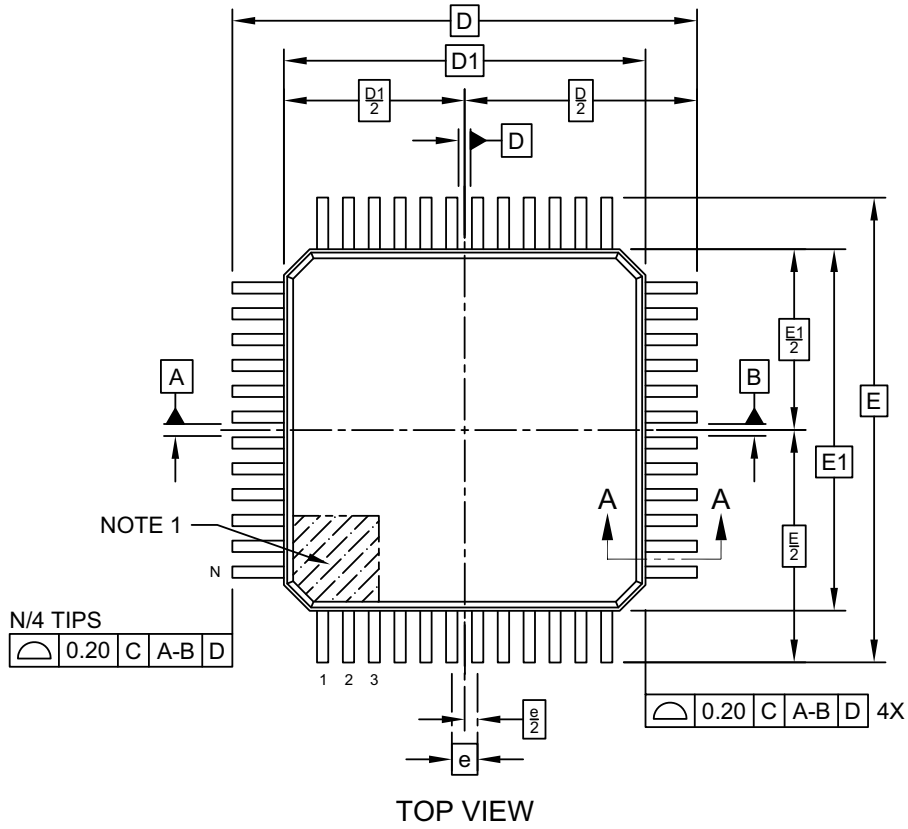
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2085B Sheet 1 of 1

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Plastic Thin Quad Flatpack (Y8X) - 7x7x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

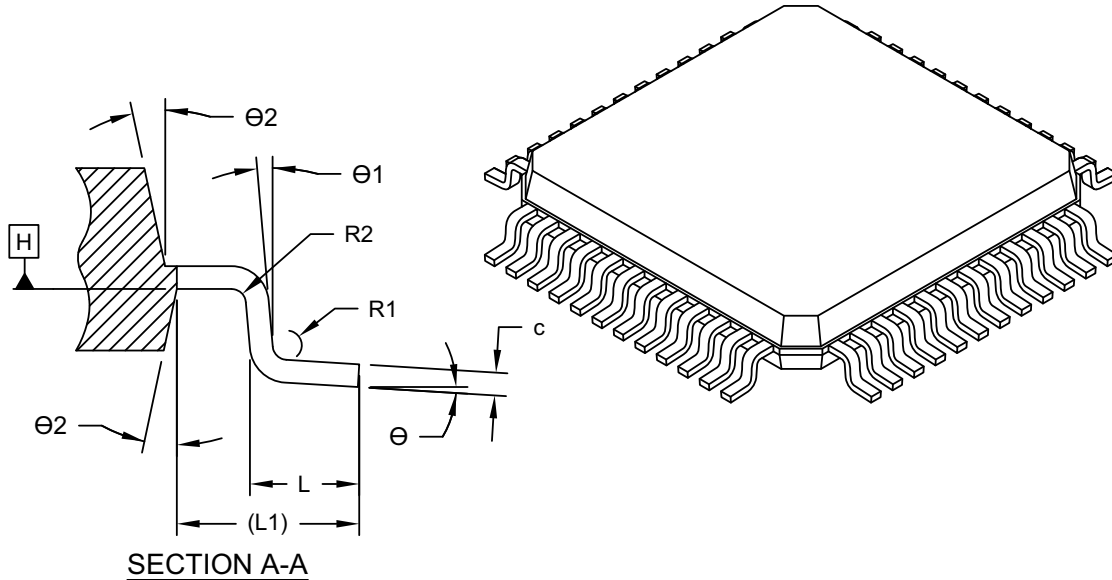


Microchip Technology Drawing C04-300-Y8 Rev B Sheet 1 of 2

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Plastic Thin Quad Flatpack (Y8X) - 7x7x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension                | Units      | MILLIMETERS |      |      |
|--------------------------|------------|-------------|------|------|
|                          |            | MIN         | NOM  | MAX  |
| Number of Terminals      | N          | 48          |      |      |
| Pitch                    | e          | 0.50 BSC    |      |      |
| Overall Height           | A          | -           | -    | 1.20 |
| Standoff                 | A1         | 0.05        | -    | 0.15 |
| Molded Package Thickness | A2         | 0.95        | 1.00 | 1.05 |
| Overall Length           | D          | 9.00 BSC    |      |      |
| Molded Package Length    | D1         | 7.00 BSC    |      |      |
| Overall Width            | E          | 9.00 BSC    |      |      |
| Molded Package Width     | E1         | 7.00 BSC    |      |      |
| Terminal Width           | b          | 0.17        | 0.22 | 0.27 |
| Terminal Thickness       | c          | 0.09        | -    | 0.16 |
| Terminal Length          | L          | 0.45        | 0.60 | 0.75 |
| Footprint                | L1         | 1.00 REF    |      |      |
| Lead Bend Radius         | R1         | 0.08        | -    | -    |
| Lead Bend Radius         | R2         | 0.08        | -    | 0.20 |
| Foot Angle               | $\theta$   | 0°          | 3.5° | 7°   |
| Lead Angle               | $\theta_1$ | 0°          | -    | -    |
| Terminal-to-Exposed-Pad  | $\theta_2$ | 11°         | 12°  | 13°  |

**Notes:**

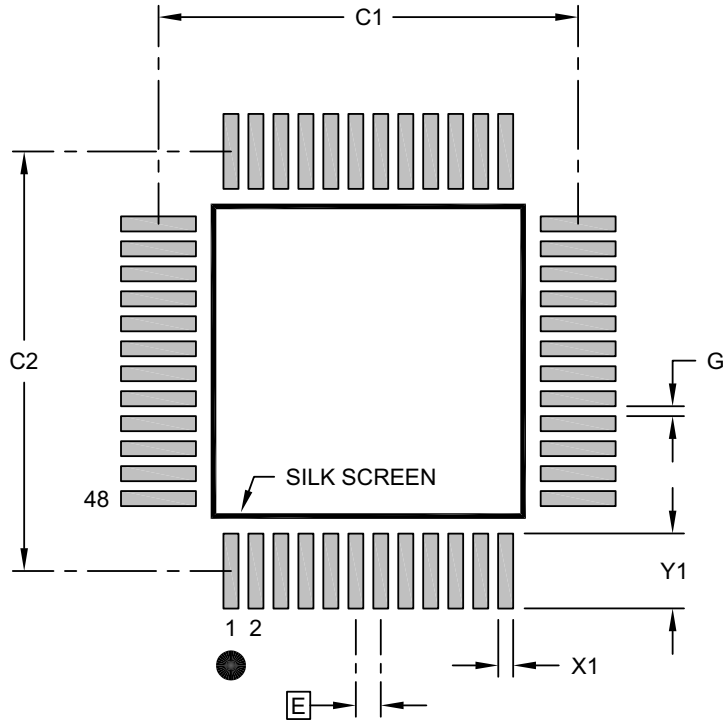
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Dimensioning and tolerancing per ASME Y14.5M  
 BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
 REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-300-Y8 Rev B Sheet 2 of 2

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Plastic Thin Quad Flatpack (Y8X) - 7x7x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



### RECOMMENDED LAND PATTERN

| Dimension Limits         | Units | MILLIMETERS |      |      |
|--------------------------|-------|-------------|------|------|
|                          |       | MIN         | NOM  | MAX  |
| Contact Pitch            | E     | 0.50 BSC    |      |      |
| Contact Pad Spacing      | C1    |             | 8.40 |      |
| Contact Pad Spacing      | C2    |             | 8.40 |      |
| Contact Pad Width (X48)  | X1    |             |      | 0.30 |
| Contact Pad Length (X48) | Y1    |             |      | 1.50 |
| Distance Between Pads    | G     | 0.20        |      |      |

**Notes:**

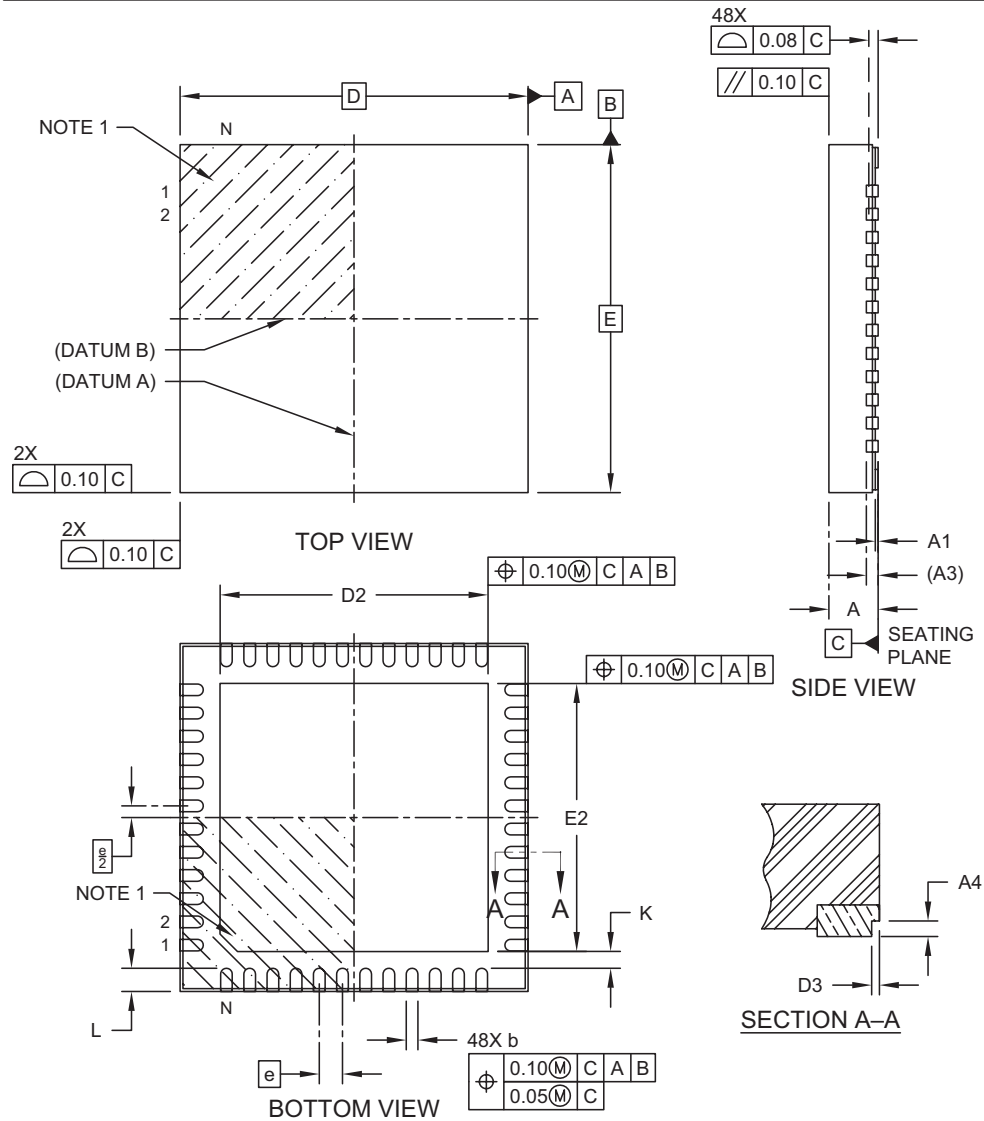
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
- For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2300-Y8 Rev B

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Very Thin Plastic Quad Flat, No Lead Package (7MX) - 6x6x0.9 mm Body [VQFN] With 4.62 mm Exposed Pad and Stepped Wettable Flanks

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



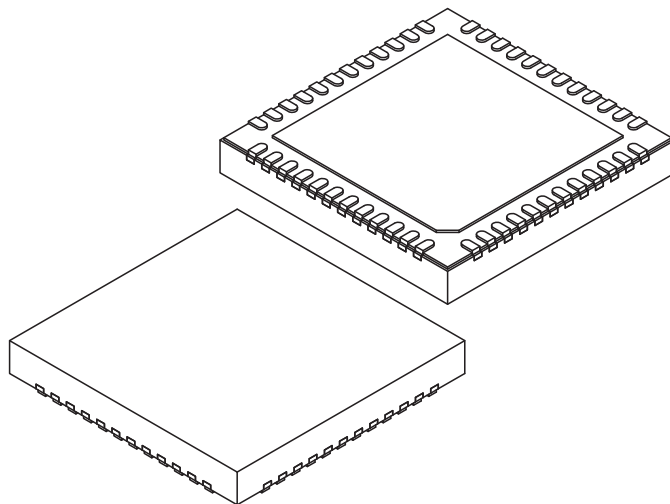
Microchip Technology Drawing C04-506 Rev A Sheet 1 of 2

© 2019 Microchip Technology Inc.

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Very Thin Plastic Quad Flat, No Lead Package (7MX) - 6x6x0.9 mm Body [VQFN] With 4.62 mm Exposed Pad and Stepped Wettable Flanks

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension Limits           | Units | MILLIMETERS |      |       |
|----------------------------|-------|-------------|------|-------|
|                            |       | MIN         | NOM  | MAX   |
| Number of Terminals        | N     | 48          |      |       |
| Pitch                      | e     | 0.40 BSC    |      |       |
| Overall Height             | A     | 0.80        | 0.85 | 0.90  |
| Standoff                   | A1    | 0.00        | 0.03 | 0.05  |
| Terminal Thickness         | A3    | 0.20 REF    |      |       |
| Overall Length             | D     | 6.00 BSC    |      |       |
| Exposed Pad Length         | D2    | 4.52        | 4.62 | 4.72  |
| Overall Width              | E     | 6.00 BSC    |      |       |
| Exposed Pad Width          | E2    | 4.52        | 4.62 | 4.72  |
| Terminal Width             | b     | 0.15        | 0.20 | 0.25  |
| Terminal Length            | L     | 0.30        | 0.40 | 0.50  |
| Terminal-to-Exposed-Pad    | K     | 0.20        | -    | -     |
| Wettable Flank Step Length | D3    | -           | -    | 0.085 |
| Wettable Flank Step Height | A4    | 0.10        | -    | 0.19  |

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.

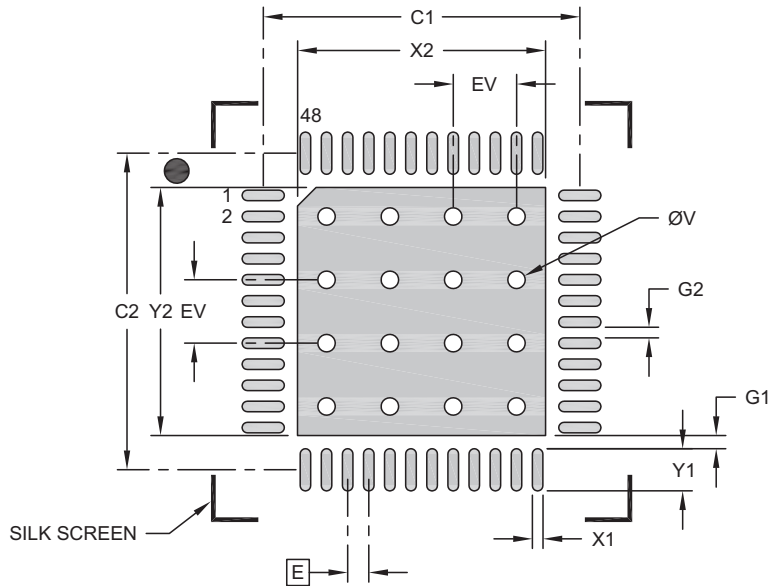
Microchip Technology Drawing C04-506 Rev A Sheet 2 of 2

© 2019 Microchip Technology Inc.

# PIC32MK GPG/MCJ with CAN FD Family

## 48-Lead Very Thin Plastic Quad Flat, No Lead Package (7MX) - 6x6x0.9 mm Body [VQFN] With 4.62 mm Exposed Pad and Stepped Wettable Flanks

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



### RECOMMENDED LAND PATTERN

| Dimension Limits                 | Units | MILLIMETERS |      |      |
|----------------------------------|-------|-------------|------|------|
|                                  |       | MIN         | NOM  | MAX  |
| Contact Pitch                    | E     | 0.40 BSC    |      |      |
| Optional Center Pad Width        | X2    |             |      | 4.70 |
| Optional Center Pad Length       | Y2    |             |      | 4.70 |
| Contact Pad Spacing              | C1    |             | 6.00 |      |
| Contact Pad Spacing              | C2    |             | 6.00 |      |
| Contact Pad Width (X48)          | X1    |             |      | 0.20 |
| Contact Pad Length (X48)         | Y1    |             |      | 0.80 |
| Contact Pad to Center Pad (X48)  | G1    | 0.25        |      |      |
| Contact Pad to Contact Pad (X44) | G2    | 0.20        |      |      |
| Thermal Via Diameter             | V     |             | 0.33 |      |
| Thermal Via Pitch                | EV    |             | 1.20 |      |

**Notes:**

- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
- For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

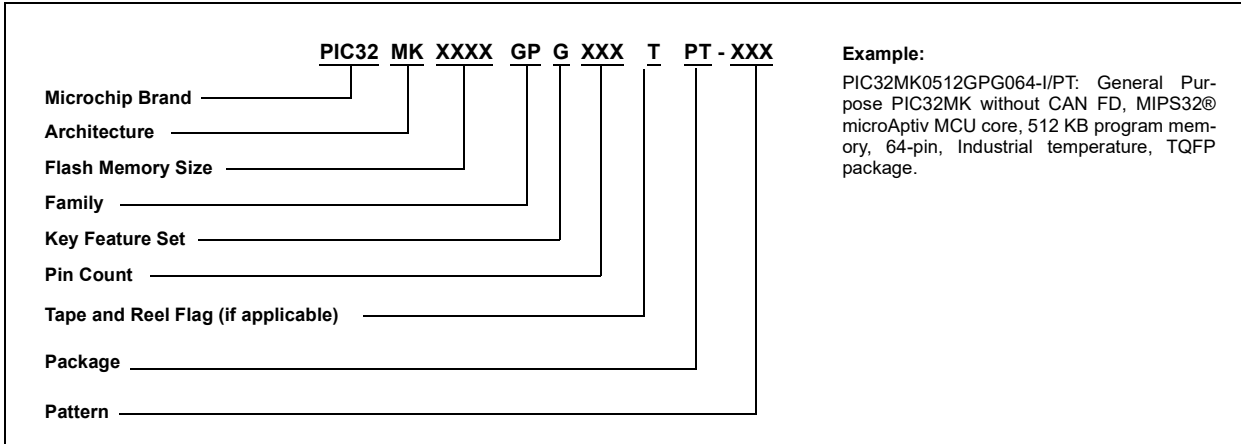
Microchip Technology Drawing C04-2506 Rev A

© 2019 Microchip Technology Inc.

# PIC32MK GPG/MCJ with CAN FD Family

## PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



**Example:**  
 PIC32MK0512GPG064-I/PT: General Purpose PIC32MK without CAN FD, MIPS32® microAptiv MCU core, 512 KB program memory, 64-pin, Industrial temperature, TQFP package.

### Flash Memory Family

|                   |  |
|-------------------|--|
| Architecture      | MK = MIPS32® microAptiv MCU Core with Floating Point Unit (FPU)  |
| Flash Memory Size | 0512 = 512 KB<br>0256 = 256 KB<br>0128 = 128 KB  |
| Family            | GP = General Purpose Microcontroller Family<br>MC = Motor Control Microcontroller Family   |
| Key Feature       | G = PIC32 GP Family Features (without CAN)<br>J = PIC32 MC Family Features (with CAN FD, PWM, and QEI)   |
| Pin Count         | 064 = 64-pin<br>048 = 48-pin   |
| Package           | R4X = 64-Lead (9x9x0.9 mm) QFN (Quad Flat-pack)<br>PT = 64-Lead (10x10x1 mm) TQFP (Thin Quad Flat-pack)<br>7MX = 48-Lead (6x6x0.8 mm) VQFN (Very Thin Quad Flat-pack)<br>Y8X = 48-Lead (7x7x1 mm) TQFP (Thin Quad Flat-pack) |
| Pattern           | Three-digit QTP, SQTP, Code or Special Requirements (blank otherwise)<br>ES = Engineering Sample   |



# PIC32MK GPG/MCJ with CAN FD Family

## APPENDIX A: REVISION HISTORY

### Revision A (May 2019)

This is the initial released version of the document.

### Revision B (March 2020)

This revision included typographical changes throughout the document. The letter designation “GPH” was removed from the document.

The following updates were made for this revision of the document:

| Section   | Updates   |
|---|---|
| <b>Section “32-bit General Purpose and Motor Control Application MCUs with CAN FD, FPU, ECC Flash, and up to 512 KB Flash, 64 KB SRAM, and Op amps”</b> | <ul style="list-style-type: none"><li>Table 1, “PIC32MK Motor Control and General Purpose (MC and GP) Family Features,” on page 3 was updated to remove GPH references and UQFN was corrected to VQFN</li><li>Table 2, “Pin Names for 64-pin General Purpose (GPG) Devices,” on page 4, Table 4, “Pin Names for 48-pin General Purpose (GPG) Devices,” on page 6, and Table 5, “Pin Names for 48-pin Motor Control (MCJ) Devices,” on page 7 were updated to remove GPH references and UQFN was corrected to VQFN</li></ul> |
| <b>Section 1.0 “Device Overview”</b>  | <ul style="list-style-type: none"><li>Updated all tables to remove GPH references and corrected UQFN to VQFN</li></ul>  |
| <b>Section 2.0 “Guidelines for Getting Started with 32-bit MCUs”</b>  | <ul style="list-style-type: none"><li>Updated the note for <b>FIGURE 2-1: “Recommended Minimum Connection”</b> to remove the reference to Aluminum or electrolytic capacitors</li></ul>   |
| <b>Section 6.0 “CPU Exceptions and Interrupt Controller”</b>  | <ul style="list-style-type: none"><li>Updated Table 6-1, “ISR Latency Information,” on page 91 with new values under the Comment column</li><li>Updated Table 6-4, “Interrupt Register Map,” on page 106</li></ul>  |
| <b>Section 10.0 “I/O Ports”</b>   | <ul style="list-style-type: none"><li>Updated the Notes for Table 10-3, “PORTA Register Map,” on page 195, Table 10-4, “PORTB Register Map,” on page 197, Table 10-5, “PORTC Register Map,” on page 198, Table 10-6, “PORTD Register Map,” on page 199, Table 10-7, “PORTE Register Map,” on page 201, Table 10-8, “PORTf Register Map,” on page 203, Table 10-9, “PORTG Register Map,” on page 204, and Table 10-10, “Peripheral Pin Select Input Register Map,” on page 205</li></ul>                                     |
| <b>Section 23.0 “12-bit High-Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC)”</b>                                       | <ul style="list-style-type: none"><li>Updated the WOVERR bit of Register 23-28 ADCDSTAT: ADC DMA Status Register with new naming and description</li></ul>  |

# PIC32MK GPG/MCJ with CAN FD Family

| Section                                   | Updates  |
|---|--|
| Section 25.0 “Op Amp/Comparator Module”   | <ul style="list-style-type: none"> <li>Updated <a href="#">FIGURE 25-9: “OP-AMP GAIN ≥ 2: HIGH PERFORMANCE ≥ 10 MHz SINGLE ENDED MODE REQUIREMENTS (CMxCON&lt;OPLPWR&gt; = 0)”</a>, <a href="#">FIGURE 25-10: “OP-AMP HIGH PERFORMANCE ≥ 10 MHz DIFFERENTIAL MODE REQUIREMENTS (CMxCON&lt;OPLPWR&gt; = 0)”</a>, <a href="#">FIGURE 25-12: “OP-AMP GAIN ≥ 2: LOW POWER MODE ≤10MHZ REQUIREMENTS (CMxCON&lt;OPLPWR&gt; = 1)”</a>, and <a href="#">FIGURE 25-13: “OP AMP LOW POWER DIFFERENTIAL MODE ≤10MHZ REQUIREMENTS (CMxCON&lt;OPLPWR&gt; = 1)”</a> with additional text</li> <li>Updated <a href="#">FIGURE 25-14: “Op amp Configuration circuit examples”</a>, <a href="#">FIGURE 25-15: “OP AMP Configuration Circuit ExAmPles”</a>, <a href="#">FIGURE 25-16: “Op amp Configuration circuit examples”</a>, <a href="#">FIGURE 25-17: “Op amp Configuration circuit examples”</a>, and <a href="#">FIGURE 25-18: “Op amp Configuration circuit examples”</a> with new diagrams</li> </ul> |
| Section 29.0 “Motor Control PWM Module”   | <ul style="list-style-type: none"> <li>Updated Register 29-16 DTRx: PWM Dead Time Register ‘x’ (‘x’ = 1 through 12), Register 29-17 ALTDTRx: PWM Alternate Dead Time Register ‘x’ (‘x’ = 1 through 12), and Register 29-23 LEBCONx: Leading-Edge Blanking Control Register ‘x’ (‘x’ = 1 through 12) with the addition of a note</li> <li>Added Figure <a href="#">FIGURE 29-2: “3 Phase AC Motor Control Example”</a></li> </ul>   |
| Section 36.0 “Electrical Characteristics” | <ul style="list-style-type: none"> <li>Updated <a href="#">Table 36-3, “Thermal Packaging Characteristics,”</a> on <a href="#">page 596</a> with new VQFN packaging information</li> <li>Updated <a href="#">Table 36-28, “Op Amp Specifications,”</a> on <a href="#">page 619</a> with a new value for the OA16 parameter</li> <li>Added <a href="#">Table 36-13, “DC CHARACTERISTICS: PROGRAM FLASH MEMORY WAIT STATES FOR ACTIVE HIGH POWER MODE,”</a> on <a href="#">page 605</a></li> </ul>   |
| Section 38.0 “Packaging Information”      | <ul style="list-style-type: none"> <li>Updated UQFN packaging to VQFN packaging with new diagrams for the VQFN 48 pin package</li> </ul>   |

## Revision C (07/2020)

This revision included typographical changes throughout the document. The following updates were made for this revision of the document:

| Section                         | Updates   |
|---------------------------------|---|
| 4.0 “Flash Program Memory”      | <ul style="list-style-type: none"> <li>Updated <a href="#">Register 4-8 NVMCON2: Flash Programming Control Register 2</a> with new Reserved bits</li> </ul>                               |
| 8.0 “Prefetch Module”           | <ul style="list-style-type: none"> <li>Updated the table for the PFMWS bit for <a href="#">Register 8-1 CHECON: Cache Module Control Register</a></li> </ul>                              |
| 25.0 “Op Amp/Comparator Module” | <ul style="list-style-type: none"> <li>Added a new section, <a href="#">Section 25.8, Op-amp Circuit Examples</a></li> <li>Added a second note to <a href="#">Figure 25-18</a></li> </ul> |

# PIC32MK GPG/MCJ with CAN FD Family

| Section                             | Updates   |
|-------------------------------------|---|
| 36.0 “Electrical Characteristics”   | <ul style="list-style-type: none"><li>• Added a new values for SDAx and SCLx in <a href="#">TABLE 36-9: “DC Characteristics: I/O Pin Input Specifications”</a></li><li>• Updated the min and max specifications for <a href="#">TABLE 36-5: “Electrical Characteristics: BOR”</a>, <a href="#">TABLE 36-6: “DC Characteristics: Operating Current (IDD Run Current with Peripheral Clocks Enabled)(1,2)”</a>, <a href="#">TABLE 36-7: “DC Characteristics: Idle Current (Iidle)”</a>, <a href="#">TABLE 36-8: “DC Characteristics: Power-Down Current (Ipd)”</a>, <a href="#">TABLE 36-12: “DC Characteristics: Program Memory(3)”</a>, <a href="#">TABLE 36-17: “Internal FRC Accuracy”</a>, <a href="#">TABLE 36-42: “Control DAC (CDAC) Specifications”</a>, <a href="#">TABLE 36-43: “CTMU Current Source Specifications”</a>, and <a href="#">Table 36-46: “Electrical Characteristics LVD”</a></li><li>• Updated <a href="#">TABLE 36-14: “Capacitive Loading Requirements on Output Pins”</a>, removed the note on the table</li><li>• Added new <a href="#">TABLE 36-19: “Internal BFRC Accuracy”</a></li></ul> |
| Appendix 38.0 Packaging Information | <ul style="list-style-type: none"><li>• <a href="#">Appendix 38.1 Package Marking Information</a></li></ul>   |

# PIC32MK GPG/MCJ with CAN FD Family

---

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at [www.microchip.com](http://www.microchip.com). This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at [www.microchip.com](http://www.microchip.com). Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the web site at: <http://microchip.com/support>**

# PIC32MK GPG/MCJ with CAN FD Family

---

---

NOTES:

# PIC32MK GPG/MCJ with CAN FD Family

---

---

---

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

---

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

### Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Klear, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTracker, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KlearNet, KlearNet logo, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICKit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2019-2020, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-5224-6408-2

For information regarding Microchip's Quality Management Systems, please visit [www.microchip.com/quality](http://www.microchip.com/quality).

# PIC32MK GPG/MCJ with CAN FD Family

## INDEX

### A

|   |     |
|---|-----|
| AC Characteristics .....                        | 606 |
| ADC Module Specifications .....                 | 644 |
| Analog-to-Digital Conversion Requirements ..... | 645 |
| Assembler                                       |     |
| MPASM Assembler .....                           | 592 |

### B

|  |                         |
|--|-------------------------|
| Block Diagrams                                   |                         |
| CLCx Input Source Selection .....                | 289                     |
| CLCx Logic Function Combinatorial Options .....  | 288                     |
| CLCx Module .....                                | 287                     |
| CPU .....  | 38                      |
| CTMU Configurations                              |                         |
| Time Measurement .....                           | 480                     |
| DMA .....  | 164                     |
| Input Capture .....                              | 248                     |
| Inter-Integrated Circuit (I2C) .....             | 268                     |
| Interrupt Controller .....                       | 102                     |
| JTAG Programming, Debugging and Trace Ports .... | 586                     |
| Op amp/Comparator Module ....                    | 452, 453, 454, 455, 456 |
| Output Compare Module .....                      | 253                     |
| PIC32 CAN Module .....                           | 389                     |
| Prefetch Module .....                            | 158                     |
| Prefetch Module Block Diagram .....              | 158                     |
| Quadrature Encoder Interface .....               | 490                     |
| Reset System .....                               | 92                      |
| RTCC .....                                       | 301                     |
| SPI Module .....                                 | 257                     |
| Timer1 .....                                     | 225                     |
| Timer2/3/4/5 (16-Bit) .....                      | 230, 231                |
| Typical Multiplexed Port Structure .....         | 192                     |
| UART .....                                       | 275                     |
| WDT and Power-up Timer .....                     | 244                     |
| Brown-out Reset (BOR)                            |                         |
| and On-Chip Voltage Regulator .....              | 586                     |

### C

|  |        |
|--|--------|
| C Compilers                                    |        |
| MPLAB XC32 .....                               | 592    |
| Charge Time Measurement Unit. <i>See</i> CTMU. |        |
| CLC  |        |
| Control Registers .....                        | 290    |
| Comparator                                     |        |
| Specifications .....                           | 610    |
| Comparator Module .....                        | 451    |
| Configurable Logic Cell (CLC) .....            | 287    |
| Configurable Logic Cell. <i>See</i> CLC.       |        |
| Configuration Bit .....                        | 565    |
| Configuring Analog Port Pins .....             | 193    |
| Controller Area Network (CAN) .....            | 388    |
| CPU  |        |
| Architecture Overview .....                    | 39     |
| Coprocesor 0 Registers .....                   | 40     |
| Core Exception Types .....                     | 103    |
| EJTAG Debug Support .....                      | 43     |
| Power Management .....                         | 43     |
| CPU Module .....                               | 26, 37 |
| CTMU   |        |
| Registers .....                                | 482    |
| Customer Change Notification Service .....     | 674    |
| Customer Notification Service .....            | 674    |
| Customer Support .....                         | 674    |

### D

|  |          |
|--|----------|
| DC Characteristics .....                     | 596      |
| I/O Pin Input Specifications .....           | 601, 602 |
| I/O Pin Output Specifications .....          | 603      |
| Idle Current (IDLE) .....                    | 599      |
| Power-Down Current (IPD) .....               | 600      |
| Program Memory .....                         | 605      |
| Temperature and Voltage Specifications ..... | 597      |
| Development Support .....                    | 591      |
| Direct Memory Access (DMA) Controller .....  | 164      |

### E

|   |     |
|---|-----|
| EJTAG Timing Requirements .....           | 651 |
| Electrical Characteristics .....          | 595 |
| AC .....                                  | 606 |
| External Clock                            |     |
| Timer1 Timing Requirements .....          | 615 |
| Timer2, 3, 4, 5 Timing Requirements ..... | 616 |
| Timing Requirements .....                 | 607 |

### F

|                            |        |
|----------------------------|--------|
| Flash Program Memory ..... | 81, 92 |
| RTSP Operation .....       | 81     |

### I

|                                      |          |
|--------------------------------------|----------|
| I/O Ports .....                      | 192      |
| Parallel I/O (PIO) .....             | 193      |
| Write/Read Timing .....              | 193      |
| Input Change Notification .....      | 193      |
| Instruction Set .....                | 588, 586 |
| Inter-Integrated Circuit (I2C) ..... | 267      |
| Internal FRC Accuracy .....          | 609      |
| Internal LPRC Accuracy .....         | 609      |
| Internet Address .....               | 674      |
| Interrupt Controller                 |          |
| IRG, Vector and Bit Location .....   | 106      |

### M

|  |     |
|--|-----|
| Memory Maps  |     |
| Devices with 1024 KB Program Memory and 512 KB RAM ..... | 61  |
| Devices with 512 KB Program Memory .....                 | 60  |
| Memory Organization .....                                | 59  |
| Layout .....   | 59  |
| Microchip Internet Web Site .....                        | 674 |
| Motor Control PWM .....                                  | 505 |
| MPLAB ASM30 Assembler, Linker, Librarian .....           | 592 |
| MPLAB ICD 3 In-Circuit Debugger .....                    | 593 |
| MPLAB PM3 Device Programmer .....                        | 593 |
| MPLAB REAL ICE In-Circuit Emulator System .....          | 593 |
| MPLAB X Integrated Development Environment Software .... | 591 |
| MPLINK Object Linker/MPLIB Object Librarian .....        | 592 |

### O

|                                |     |
|--------------------------------|-----|
| Op Amp                         |     |
| Specifications .....           | 618 |
| Oscillator Configuration ..... | 140 |
| Output Compare .....           | 253 |

### P

|                 |     |
|-----------------|-----|
| Packaging ..... | 654 |
| Details .....   | 655 |
| Marking .....   | 654 |



# PIC32MK GPG/MCJ with CAN FD Family

|   |              |  |     |
|---|--------------|--|-----|
| Parallel Slave Port Requirements .....  | 649          | ADCCMPxCON (ADC Digital Comparator 'x' Control Register ('x' = 2 through 4)) ..... | 368 |
| PICkit 3 In-Circuit Debugger/Programmer .....                                       | 593          | ADCCNTB (ADC Channel Sample Count Base Address) .....                              | 373 |
| Pinout I/O Descriptions (table) .....   | 10           | ADCCON1 (ADC Control Register 1) .....   | 322 |
| PORTB Register Map (64-pin and 100-pin Devices) .....                               | 208          | ADCCON2 (ADC Control Register 2) .....   | 325 |
| Power-on Reset (POR) .....  |              | ADCCON3 (ADC Control Register 3) .....   | 328 |
| and On-Chip Voltage Regulator .....   | 586          | ADCCSS1 (ADC Common Scan Select Register 1) .....                                  | 344 |
| Power-Saving Features .....   | 558          | ADCCSS2 (ADC Common Scan Select Register 2) .....                                  | 345 |
| with CPU Running .....  | 558          | ADCDATAx (ADC Output Data Register ('x' = 0-27, 33-41, and 45-53)) .....           | 374 |
| Prefetch Module .....   | 158          | ADCDMAB (ADC Channel Sample count Base Address) .....                              | 373 |
| <b>Q</b>  |              | ADCDSTAT1 (ADC Data Ready Status Register 1) .....                                 | 346 |
| Quadrature Encoder Interface (QEI) .....  | 489          | ADCDSTAT2 (ADC Data Ready Status Register 2) .....                                 | 346 |
| <b>R</b>  |              | ADCEIEN1 (ADC Early Interrupt Enable Register 1) .....                             | 378 |
| Real-Time Clock and Calendar (RTCC) .....   | 301          | ADCEIEN2 (ADC Early Interrupt Enable Register 2) .....                             | 379 |
| Register Map .....  |              | ADCEIEN2 (ADC Early Interrupt Status Register 2) ..                                | 381 |
| Comparator Voltage Reference .....  | 555          | ADCFLTRx (ADC Digital Filter 'x' Register ('x' = 1 through 6)) .....               | 349 |
| CTMU .....  | 473, 481     | ADCGIRQEN1 (ADC Interrupt Enable Register 1) ...                                   | 342 |
| Device ADC Calibration Summary .....  | 569          | ADCIMCON1 (ADC Input Mode Control Register 1) .....                                | 334 |
| Device Configuration Word Summary .....   | 566, 567     | ADCIMCON2 (ADC Input Mode Control Register 2) .....                                | 337 |
| Device EEDATA Calibration Summary .....   | 569          | ADCIMCON3 (ADC Input Mode Control Register 3) .....                                | 339 |
| Device Serial Number Summary .....  | 570          | ADCIMCON4 (ADC Input Mode Control Register 4) .....                                | 341 |
| DMA Channel 0-3 .....   | 166          | ADCIRQEN2 (ADC Interrupt Enable Register 2) .....                                  | 343 |
| DMA CRC .....   | 165          | ADCSYSCFG0 (ADC System Configuration Register 0) ..                                | 385 |
| DMA Global .....  | 165          | ADCSYSCFG1 (ADC System Configuration Register 1) ..                                | 386 |
| Flash Controller .....  | 82, 237, 245 | ADCTRG1 (ADC Trigger Source 1 Register) .....                                      | 351 |
| I2C1 and I2C2 .....   | 269          | ADCTRG2 (ADC Trigger Source 2 Register) .....                                      | 353 |
| Input Capture 1-9 .....   | 250          | ADCTRG3 (ADC Trigger Source 3 Register) .....                                      | 355 |
| Interrupt .....   | 116          | ADCTRG4 (ADC Trigger Source 4 Register) .....                                      | 357 |
| Op amp/Comparator .....   | 473          | ADCTRG5 (ADC Trigger Source 5 Register) .....                                      | 359 |
| Oscillator Configuration .....  | 144          | ADCTRG6 (ADC Trigger Source 6 Register) .....                                      | 361 |
| Peripheral Pin Select Input .....   | 215          | ADCTRG7 (ADC Trigger Source 7 Register) .....                                      | 363 |
| Peripheral Pin Select Output .....  | 219          | ADCTRGMODE (ADC Triggering Mode for Dedicated ADC) .....                           | 332 |
| PORTA (100-pin Devices) .....   | 205          | ADCTRGSNS (ADC Trigger Level/Edge Sensitivity) .....                               | 375 |
| PORTA (64-pin Devices) .....  | 207          | ADCxCFG (ADCx Configuration Register 'x' ('x' = 0 through 5 and 7)) .....          | 384 |
| PORTB .....   | 208          | ADCxTIME (Dedicated ADCx Timing Register 'x' ('x' = 0 through 5)) .....            | 376 |
| PORTC (64-pin and 100-pin Devices) .....  | 209          | ALRMDATE (Alarm Date Value) .....  | 310 |
| PORTD .....   | 213          | ALRMDATECLR (ALRMDATE Clear) .....   | 310 |
| PORTD (100-pin Devices) .....   | 211          | ALRMDATESET (ALRMDATE Set) .....   | 310 |
| PORTE (100-pin Devices) .....   | 214          | ALRMTIME (Alarm Time Value) .....  | 309 |
| Prefetch .....  | 159          | ALRMTIMECLR (ALRMTIME Clear) .....   | 310 |
| RTCC .....  | 302          | ALRMTIMEINV (ALRMTIME Invert) .....  | 310 |
| SPI1 and SPI2 .....   | 258          | ALRMTIMESET (ALRMTIME Set) .....   | 310 |
| System Bus .....  | 68           | ALTDTRx (PWM Alternate Dead Time Register) .....                                   | 542 |
| System Bus Target 0 .....   | 68           | ALTDTRx (PWM Alternate Dead-Time Register) .....                                   | 543 |
| System Bus Target 1 .....   | 69           | AUXCONx (PWM Auxiliary Control Register) .....                                     | 551 |
| System Bus Target 2 .....   | 71           | CHECON (Cache Module Control) .....  | 160 |
| System Bus Target 3 .....   | 72           | CHEHIT (Cache Hit Status) .....  | 162 |
| System Control .....  | 93           | CHEMIS (Cache Miss Status) .....   | 163 |
| Timer1-Timer9 .....   | 226, 232     | CHOP (PWM Chop Clock Generator Register) .....                                     | 527 |
| UART1 and UART2 .....   | 276          | CLCxCON (CLCx Control) .....   | 292 |
| Register Maps .....   |              | CLCxGLS (CLCx Gate Logic Input Select) .....                                       | 298 |
| CLC1, CLC2 and CLC3 .....   | 291          | CLCxSEL (CLCx Input MUX Select) .....  | 294 |
| Registers .....   |              | CMSTAT (Op amp/Comparator Status) .....  | 474 |
| [ <i>pin name</i> ]R (Peripheral Pin Select Input) .....                            | 222          | CMxCON (Op amp/Comparator 'x' Control) .....                                       | 475 |
| AD1CON1 (A/D Control 1) .....   | 310          | CMxMSKCON (Op amp/Comparator 'x' Mask Control) ..                                  | 478 |
| AD1CON1 (ADC Control 1) .....   | 310          |  |     |
| ADCANCON (ADC Analog Warm-up Control Register) ..                                   | 382          |  |     |
| ADCBASE (ADC Base) .....  | 370          |  |     |
| ADCCMP1CON (ADC Digital Comparator 1 Control Register) .....                        | 365          |  |     |
| ADCCMPENx (ADC Digital Comparator 'x' Enable Register ('x' = 1 through 4)) .....    | 347          |  |     |
| ADCCMPx (ADC Digital Comparator 'x' Limit Value Register ('x' = 1 through 4)) ..... | 348          |  |     |

# PIC32MK GPG/MCJ with CAN FD Family

|  |        |  |               |
|--|--------|--|---------------|
| CNCONx (Change Notice Control for PORTx) .....                                     | 223    | LEBCONx (Leading Edge Blanking Control Register)....               | 544, 548, 549 |
| CONFIG (Configuration Register - CP0 Register 16, Select 0) .....                  | 45     | LEBDLYx (Leading-Edge Blanking Delay Register) .                   | 550           |
| CONFIG1 (Configuration Register 1 - CP0 Register 16, Select 1) .....               | 47     | NVMADDR (Flash Address) .....                                      | 85            |
| CONFIG3 (Configuration Register 3 - CP0 Register 16, Select 3) .....               | 48     | NVMBWP (Flash Boot (Page) Write-protect) .....                     | 88            |
| CONFIG5 (Configuration Register 5 - CP0 Register 16, Select 5) .....               | 50, 51 | NVMCON (Programming Control) .....                                 | 83, 89        |
| CONFIG7 (Configuration Register 7 - CP0 Register 16, Select 7) .....               | 51     | NVMDATA (Flash Data) .....   | 86            |
| CTMUCON (CTMU Control) .....   | 482    | NVMKEY (Programming Unlock) .....                                  | 85            |
| DCHxCON (DMA Channel 'x' Control) .....  | 179    | NVMPWP (Program Flash Write-Protect) .....                         | 87            |
| DCHxCPTR (DMA Channel x Cell Pointer) .....  | 190    | NVMSRCADDR (Source Data Address) .....                             | 86            |
| DCHxCSIZ (DMA Channel x Cell-Size) .....   | 189    | OCxCON (Output Compare x Control) .....                            | 255           |
| DCHxDAT (DMA Channel x Pattern Data) .....   | 191    | OSCCON (Oscillator Control) .....                                  | 146           |
| DCHxDPTR (Channel x Destination Pointer) .....                                     | 188    | OSCTUN (FRC Tuning) .....  | 148           |
| DCHxDSA (DMA Channel x Destination Start Address) .....                            | 184    | PDCx (PWM Generator Duty Cycle Register) .....                     | 539           |
| DCHxDSIZ (DMA Channel x Destination Size) .....                                    | 186    | PHASEx (PWM Primary Phase Shift Register) .....                    | 541           |
| DCHxECON (DMA Channel x Event Control) .....                                       | 181    | PMTMR (Primary Master Time Base Timer Register) ....               | 523           |
| DCHxINT (DMA Channel x Interrupt Control) .....                                    | 182    | POSxCNT (Position Counter Register) .....                          | 500           |
| DCHxSPTR (DMA Channel x Source Pointer) .....                                      | 187    | PRISS (Priority Shadow Select) .....                               | 133           |
| DCHxSSA (DMA Channel x Source Start Address) .                                     | 184    | PSCNT (Post Status Configure DMT Count Status) 241                 |               |
| DCHxSSIZ (DMA Channel x Source Size) .....   | 185    | PSINTV (Post Status Configure DMT Interval Status) ...             | 242           |
| DCRCON (DMA CRC Control) .....   | 175    | PTCON (PWM Primary Time Base Control Register) ....                | 520           |
| DCRCDATA (DMA CRC Data) .....  | 177    | PTPER (Primary Master Time Base Period Register) ....              | 522           |
| DCRCXOR (DMA CRCXOR Enable) .....  | 178    | PWMCONx (PWM Control Register) .....                               | 529           |
| DEVCFG0 (Device Configuration Word 0) .....  | 572    | PWMKEY   |               |
| DEVCFG1 (Device Configuration Word 1) .....  | 575    | (PWM Unlock Register) .....  | 528           |
| DEVCFG2 (Device Configuration Word 2) .....  | 578    | QEIXCMLP (Capture Low Register) .....                              | 504           |
| DEVCFG3 (Device Configuration Word 3) .....  | 580    | QEIXCON QEIX Control) .....  | 494           |
| DEVCP0 (Device Code-protect 0) .....   | 571    | QEIXICC (QEIX Initialize/Capture/Compare Register) ....            | 504           |
| DEVID (Device and Revision ID) .....   | 584    | QEIXIOC (QEIX I/O Control) .....                                   | 496           |
| DEVSIGN0 (Device Signature Word 0) .....   | 571    | QEIXSTAT (QEIX Status) .....                                       | 498           |
| DMAADDR (DMA Address) .....  | 174    | REFOxCON (Reference Oscillator Control ('x' = 1-4)) ...            | 151           |
| DMAADDR (DMR Address) .....  | 174    | REFOxTRIM (Reference Oscillator Trim ('x' = 1-4)) .                | 153           |
| DMACON (DMA Controller Control) .....  | 172    | RPnR (Peripheral Pin Select Output) .....                          | 222           |
| DMASTAT (DMA Status) .....   | 173    | RSWRST (Software Reset) .....                                      | 96, 97, 99    |
| DMSTAT (Deadman Timer Status) .....  | 240    | RTCCON (RTCC Control) .....  | 303           |
| DMTCLR (Deadman Timer Clear) .....   | 239    | RTCDATE (RTC Date Value) .....                                     | 308           |
| DMTCNT (Deadman Timer Count) .....   | 241    | RTCTIME (RTC Time Value) .....                                     | 307           |
| DMTCON (Deadman Timer Control) .....   | 238    | SBFLAG (System Bus Status Flag) .....                              | 73            |
| DMTPRECLR (Deadman Timer Preclear) .....   | 238    | SBTxECLRM (System Bus Target 'x' Multiple Error Clear              | 77            |
| FCCR (Floating Point Condition Codes Register - CP1 Register 25) .....             | 53     | SBTxECLRS (System Bus Target 'x' Single Error Single)              | 77            |
| FCSR (Floating Point Control and Status Register - CP1 Register 31) .....          | 56     | SBTxECON (System Bus Target 'x' Error Control) ....                | 76            |
| FENR (Floating Point Exceptions and Modes Enable Register - CP1 Register 28) ..... | 55     | SBTxELOG1 (System Bus Target 'x' Error Log 1) ....                 | 74            |
| FEXR (Floating Point Exceptions Status Register - CP1 Register 26) .....           | 54     | SBTxELOG2 (System Bus Target 'x' Error Log 2) ....                 | 76            |
| FIR (Floating Point Implementation Register - CP1 Register 0) .....                | 52     | SBTxRDy (System Bus Target 'x' Region 'y' Read Permissions) .....  | 79            |
| I2CxCON (I2C Control) .....  | 270    | SBTxREGy (System Bus Target 'x' Region 'y') .....                  | 78            |
| I2CxSTAT (I2C Status) .....  | 272    | SBTxWRy (System Bus Target 'x' Region 'y' Write Permissions) ..... | 80            |
| ICxCON (Input Capture x Control) .....   | 251    | SDCx (PWM Secondary Duty Cycle Register) .....                     | 540           |
| IFSx (Interrupt Flag Status) .....   | 136    | SEVTCMP (Special Event Compare Register) .....                     | 523           |
| INDxCNT (Index Counter Register) .....   | 503    | SMTMR (Secondary Master Time Base Timer Register)                  | 526           |
| INTCON (Interrupt Control) .....   | 132    | SPIxBRG (SPIx Baud Rate Generator) .....                           | 265           |
| INTSTAT (Interrupt Status) .....   | 135    | SPIxBUF (SPIx Buffer) .....  | 265           |
| INTxHLD (Interval Timer Hold Register) .....                                       | 502    | SPIxCON (SPI Control) .....  | 259           |
| INTxTMR (Interval Timer Register) .....  | 503    | SPIxCON2 (SPI Control 2) .....                                     | 262           |
| IOCONx (PWM I/O Control Register) .....  | 532    | SPIxSTAT (SPI Status) .....  | 263           |
| IPCx (Interrupt Priority Control) .....  | 137    | SPLLCON (System PLL Control) .....                                 | 149           |
| IPTMR Interrupt Proximity Timer) .....   | 135    |  |               |

# PIC32MK GPG/MCJ with CAN FD Family

|   |     |   |          |
|---|-----|---|----------|
| SSEVTCMP (PWM Secondary Special Event Compare Register) ..... | 525 | Input Capture (CAPx) .....                    | 616      |
| STCON (Secondary Master Time Base Control Register) .....     | 524 | Motor Control PWM Fault .....                 | 649, 650 |
| STPER (Secondary Master Time Base Period Register) .....      | 525 | OCx/PWM .....                                 | 617      |
| STRIGx (Secondary PWM Trigger Compare Register) ..            | 548 | Output Compare (OCx).....                     | 617      |
| T1CON (Type A Timer Control) .....                            | 227 | QEA/QEB Input .....                           | 642      |
| TMR (PWM Timer Register) .....                                | 552 | SPIx Master Mode (CKE = 0) .....              | 621      |
| TMRx (PWM Timer Register 'x') .....                           | 552 | SPIx Master Mode (CKE = 1) .....              | 624      |
| TRGCONx (PWM Trigger Control Register) .....                  | 546 | SPIx Slave Mode (CKE = 0) .....               | 628      |
| TRIGx (PWM Trigger Compare Value Register).....               | 545 | SPIx Slave Mode (CKE = 1) .....               | 632      |
| TxCON (Type B Timer Control) .....                            | 234 | Timer1, 2, 3, 4, 5 External Clock .....       | 615      |
| VELxCNT (Velocity Counter Register).....                      | 501 | TimerQ (QEI Module) External Clock .....      | 641      |
| VELxHLD (Velocity Hold Register) .....                        | 502 | UART Reception.....                           | 285      |
| WDTCN (Watchdog Timer Control).....                           | 246 | UART Transmission (8-bit or 9-bit Data) ..... | 285      |
| Revision History .....  | 668 | Timing Requirements                           |          |
| RTCALRM (RTC ALARM Control).....                              | 305 | CLKO and I/O .....                            | 611      |
| <b>S</b>  |     | Timing Specifications                         |          |
| Serial Peripheral Interface (SPI) .....                       | 257 | CAN I/O Requirements .....                    | 643      |
| Software Simulator (MPLAB X SIM) .....                        | 593 | Input Capture Requirements.....               | 616      |
| Special Features .....  | 565 | Motor Control PWM Requirements.....           | 649      |
| <b>T</b>  |     | Output Compare Requirements.....              | 617      |
| Timer1 Module .....   | 224 | QEI External Clock Requirements .....         | 641      |
| Timer2/3, Timer4/5, Timer6/7, and Timer8/9 Modules.....       | 230 | Quadrature Decoder Requirements.....          | 642      |
| Timing Diagrams   |     | Simple OCx/PWM Mode Requirements .....        | 617, 620 |
| CAN I/O.....  | 643 | SPIx Master Mode (CKE = 0) Requirements ..... | 622      |
| EJTAG .....   | 651 | SPIx Master Mode (CKE = 1) Requirements ..... | 625      |
| External Clock.....   | 607 | SPIx Slave Mode (CKE = 1) Requirements.....   | 633      |
| I/O Characteristics .....                                     | 611 | SPIx Slave Mode Requirements (CKE = 0).....   | 629      |
| I2Cx Bus Data (Master Mode) .....                             | 636 | <b>U</b>                                      |          |
| I2Cx Bus Data (Slave Mode) .....                              | 639 | UART .....                                    | 275      |
| I2Cx Bus Start/Stop Bits (Master Mode).....                   | 636 | <b>V</b>                                      |          |
| I2Cx Bus Start/Stop Bits (Slave Mode) .....                   | 639 | Voltage Regulator (On-Chip) .....             | 586      |
|   |     | <b>W</b>                                      |          |
|   |     | WWW Address .....                             | 674      |



# MICROCHIP

## Worldwide Sales and Service

### AMERICAS

**Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
[http://www.microchip.com/  
support](http://www.microchip.com/support)  
Web Address:  
[www.microchip.com](http://www.microchip.com)

#### Atlanta

Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

#### Austin, TX

Tel: 512-257-3370

#### Boston

Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088

#### Chicago

Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075

#### Dallas

Addison, TX  
Tel: 972-818-7423  
Fax: 972-818-2924

#### Detroit

Novi, MI  
Tel: 248-848-4000

#### Houston, TX

Tel: 281-894-5983

#### Indianapolis

Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453  
Tel: 317-536-2380

#### Los Angeles

Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608  
Tel: 951-273-7800

#### Raleigh, NC

Tel: 919-844-7510

#### New York, NY

Tel: 631-435-6000

#### San Jose, CA

Tel: 408-735-9110  
Tel: 408-436-4270

#### Canada - Toronto

Tel: 905-695-1980  
Fax: 905-695-2078

### ASIA/PACIFIC

**Australia - Sydney**  
Tel: 61-2-9868-6733

**China - Beijing**  
Tel: 86-10-8569-7000

**China - Chengdu**  
Tel: 86-28-8665-5511

**China - Chongqing**  
Tel: 86-23-8980-9588

**China - Dongguan**  
Tel: 86-769-8702-9880

**China - Guangzhou**  
Tel: 86-20-8755-8029

**China - Hangzhou**  
Tel: 86-571-8792-8115

**China - Hong Kong SAR**  
Tel: 852-2943-5100

**China - Nanjing**  
Tel: 86-25-8473-2460

**China - Qingdao**  
Tel: 86-532-8502-7355

**China - Shanghai**  
Tel: 86-21-3326-8000

**China - Shenyang**  
Tel: 86-24-2334-2829

**China - Shenzhen**  
Tel: 86-755-8864-2200

**China - Suzhou**  
Tel: 86-186-6233-1526

**China - Wuhan**  
Tel: 86-27-5980-5300

**China - Xian**  
Tel: 86-29-8833-7252

**China - Xiamen**  
Tel: 86-592-2388138

**China - Zhuhai**  
Tel: 86-756-3210040

### ASIA/PACIFIC

**India - Bangalore**  
Tel: 91-80-3090-4444

**India - New Delhi**  
Tel: 91-11-4160-8631

**India - Pune**  
Tel: 91-20-4121-0141

**Japan - Osaka**  
Tel: 81-6-6152-7160

**Japan - Tokyo**  
Tel: 81-3-6880-3770

**Korea - Daegu**  
Tel: 82-53-744-4301

**Korea - Seoul**  
Tel: 82-2-554-7200

**Malaysia - Kuala Lumpur**  
Tel: 60-3-7651-7906

**Malaysia - Penang**  
Tel: 60-4-227-8870

**Philippines - Manila**  
Tel: 63-2-634-9065

**Singapore**  
Tel: 65-6334-8870

**Taiwan - Hsin Chu**  
Tel: 886-3-577-8366

**Taiwan - Kaohsiung**  
Tel: 886-7-213-7830

**Taiwan - Taipei**  
Tel: 886-2-2508-8600

**Thailand - Bangkok**  
Tel: 66-2-694-1351

**Vietnam - Ho Chi Minh**  
Tel: 84-28-5448-2100

### EUROPE

**Austria - Wels**  
Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

**Denmark - Copenhagen**  
Tel: 45-4485-5910  
Fax: 45-4485-2829

**Finland - Espoo**  
Tel: 358-9-4520-820

**France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

**Germany - Garching**  
Tel: 49-8931-9700

**Germany - Haan**  
Tel: 49-2129-3766400

**Germany - Heilbronn**  
Tel: 49-7131-72400

**Germany - Karlsruhe**  
Tel: 49-721-625370

**Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

**Germany - Rosenheim**  
Tel: 49-8031-354-560

**Israel - Ra'anana**  
Tel: 972-9-744-7705

**Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781

**Italy - Padova**  
Tel: 39-049-7625286

**Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340

**Norway - Trondheim**  
Tel: 47-7288-4388

**Poland - Warsaw**  
Tel: 48-22-3325737

**Romania - Bucharest**  
Tel: 40-21-407-87-50

**Spain - Madrid**  
Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

**Sweden - Gothenberg**  
Tel: 46-31-704-60-40

**Sweden - Stockholm**  
Tel: 46-8-5090-4654

**UK - Wokingham**  
Tel: 44-118-921-5800  
Fax: 44-118-921-5820