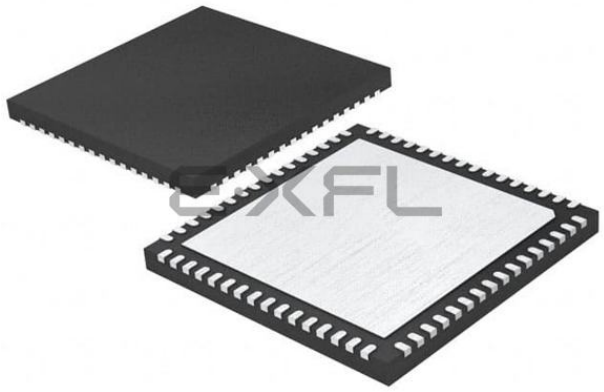
# Microchip Technology - PIC32MK0512GPD064T-E/MR Datasheet



**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | MIPS32® microAptiv™ |
| Core Size | 32-Bit Single-Core |
| Speed | 80MHz |
| Connectivity | IrDA, LINbus, PMP, SPI, UART/USART, USB OTG |
| Peripherals | Brown-out Detect/Reset, DMA, I²S, POR, WDT |
| Number of I/O | 48 |
| Program Memory Size | 512KB (512K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | 4K x 8 |
| RAM Size | 128K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.2V ~ 3.6V |
| Data Converters | A/D 26x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 64-VFQFN Exposed Pad |
| Supplier Device Package | 64-VQFN (9x9) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic32mk0512gpd064t-e-mr |

**TABLE 1-10: PMP PINOUT I/O DESCRIPTIONS**

| Pin Name | Pin Number | | Pin Type | Buffer Type | Description |
|---|---|---|---|---|---|
| | 100-pin TQFP | 64-pin QFN/ TQFP | | | |
| PMA0 | 44 | 30 | O | TTL/CMOS | Parallel Master Port Address (Demultiplexed Master mode) or Address/ Data (Multiplexed Master modes) |
| PMA1 | 43 | 29 | O | TTL/CMOS | |
| PMA2 | 14 | 8 | O | TTL/CMOS | |
| PMA3 | 12 | 6 | O | TTL/CMOS | |
| PMA4 | 11 | 5 | O | TTL/CMOS | |
| PMA5 | 10 | 4 | O | TTL/CMOS | |
| PMA6 | 29 | 16 | O | TTL/CMOS | |
| PMA7 | 28 | 22 | O | TTL/CMOS | |
| PMA8 | 50 | 32 | O | TTL/CMOS | |
| PMA9 | 49 | 31 | O | TTL/CMOS | |
| PMA10 | 42 | 28 | O | TTL/CMOS | |
| PMA11 | 41 | 27 | O | TTL/CMOS | |
| PMA12 | 35 | 24 | O | TTL/CMOS | |
| PMA13 | 34 | 23 | O | TTL/CMOS | |
| PMA14 | 71 | 45 | O | TTL/CMOS | |
| PMA15 | 70 | 44 | O | TTL/CMOS | |
| PMA16 | 77 | — | O | TTL/CMOS | |
| PMA17 | 78 | — | O | TTL/CMOS | |
| PMA18 | 91 | — | O | TTL/CMOS | |
| PMA19 | 92 | — | O | TTL/CMOS | |
| PMA20 | 95 | — | O | TTL/CMOS | |
| PMA21 | 96 | — | O | TTL/CMOS | |
| PMA22 | 97 | — | O | TTL/CMOS | |
| PMA23 | 1 | — | O | TTL/CMOS | |
| PMCS1 | 71 | 45 | O | TTL/CMOS | Parallel Master Port Chip Select 1 for PMA(13:0) |
| PMCS2 | 70 | 44 | O | TTL/CMOS | Parallel Master Port Chip Select 2 for PMA(14:0) |
| PMPRD | 82 | 53 | O | TTL/CMOS | Parallel Master Port Read Strobe |
| PMWR | 81 | 52 | O | TTL/CMOS | Parallel Master Port Write Strobe |
| PMCS1A | 97 | — | O | TTL/CMOS | Parallel Master Port Chip Select 1 for PMA(21:0) |
| PMCS2A | 1 | — | O | TTL/CMOS | Parallel Master Port Chip Select 2 for PMA(22:0) |

**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

**REGISTER 3-4: 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 |
| | — | — | — | — | — | — | — | — |

| **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 **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'

**TABLE 8-2: MIPS32® microAptiv™ MCU CORE EXCEPTION TYPES**

| Exception Type (In Order of Priority) | Description | Branches to | Status Bits Set | Debug Bits Set | EXCCODE | XC32 Function Name |
|---|---|---|---|---|---|---|
| Highest Priority | | | | | | |
| Reset | Assertion $\overline{\text{MCLR}}$ or a Power-on Reset (POR). | 0xBFC0_0000 | BEV, ERL | — | — | _on_reset |
| Soft Reset | Assertion of a software Reset. | 0xBFC0_0000 | BEV, SR, ERL | — | — | _on_reset |
| 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, ERL | — | — | _nmi_handler |
| Interrupt | Assertion of unmasked hardware or software interrupt signal. | See Table 8-3. | IPL<2:0> | — | 0x00 | See Table 8-3. |
| Deferred Watch | Deferred watch (unmasked by K\|DM=>!(K\|DM) transition). | EBASE+0x180 | WP, EXL | — | 0x17 | _general_exception_handler |
| 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 | _general_exception_handler |
| AdEL | Fetch address alignment error. Fetch reference to protected address. | EBASE+0x180 | EXL | — | 0x04 | _general_exception_handler |
| IBE | Instruction fetch bus error. | EBASE+0x180 | EXL | — | 0x06 | _general_exception_handler |
| Instruction Validity 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 0x0B | _general_exception_handler |
| Execute 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 DDBS | — | — |
| WATCH | A reference to an address that is in one of the Watch registers (data). | EBASE+0x180 | EXL | — | 0x17 | _general_exception_handler |

**REGISTER 11-12: 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

**REGISTER 11-13: 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

1111111111111111 = 65,535 byte destination size
•
•
•
0000000000000010 = 2 byte destination size
0000000000000001 = 1 byte destination size
0000000000000000 = 65,536 byte destination size

**REGISTER 12-1: UxOTGIR: USB OTG INTERRUPT STATUS REGISTER ('x' = 1 AND 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 | R/WC-0, HS | R/WC-0, HS | R/WC-0, HS | R/WC-0, HS | R/WC-0, HS | R/WC-0, HS | U-0 | R/WC-0, HS |
| | IDIF | T1MSECIF | LSTATEIF | ACTVIF | SESVDIF | SESENDIF | — | VBUSVDIF |

| **Legend:** | WC = Write '1' to clear | 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 31-8 **Unimplemented:** Read as '0'

bit 7 **IDIF:** ID State Change Indicator bit
- 1 = Change in ID state is detected
- 0 = No change in ID state is detected

bit 6 **T1MSECIF:** 1 Millisecond Timer bit
- 1 = 1 millisecond timer has expired
- 0 = 1 millisecond timer has not expired

bit 5 **LSTATEIF:** Line State Stable Indicator bit
- 1 = USB line state has been stable for 1millisecond, but different from last time
- 0 = USB line state has not been stable for 1 millisecond

bit 4 **ACTVIF:** Bus Activity Indicator bit
- 1 = Activity on the D+, D-, ID or VBUS pins has caused the device to wake-up
- 0 = Activity has not been detected

bit 3 **SESVDIF:** Session Valid Change Indicator bit
- 1 = VBUS voltage has dropped below the session end level
- 0 = VBUS voltage has not dropped below the session end level

bit 2 **SESENDIF:** B-Device VBUS Change Indicator bit
- 1 = A change on the session end input was detected
- 0 = No change on the session end input was detected

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

bit 0 **VBUSVDIF:** A-Device VBUS Change Indicator bit
- 1 = Change on the session valid input is detected
- 0 = No change on the session valid input is detected

**TABLE 20-2: SPI3 THROUGH SPI6 REGISTER MAP**

| Virtual Address (BF84_#) | 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 | |
| 7400 | SPI3CON | 31:16 | FRMEN | FRMSYNC | FRMPOL | MSSEN | 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 |
| 7410 | SPI3STAT | 31:16 | — | — | — | RXBUFELM<4:0> | | | | | — | — | — | TXBUFELM<4:0> | | | | | 0000 |
| | | 15:0 | — | — | — | FRMERR | SPIBUSY | — | — | SPITUR | SRMT | SPIROV | SPIRBE | — | SPITBE | — | SPITBF | SPIRBF | 0028 |
| 7420 | SPI3BUF | 31:16 | DATA<31:0> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | | | | | | | | | | | | | | | | | 0000 |
| 7430 | SPI3BRG | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | — | — | — | BRG<12:0> | | | | | | | | | | | | | 0000 |
| 7440 | SPI3CON2 | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | SPI SGNEXT | — | — | FRM ERREN | SPI ROVEN | SPI TUREN | IGNROV | IGNTUR | AUDEN | — | — | — | AUD MONO | — | AUDMOD<1:0> | | 0C00 |
| 7600 | SPI4CON | 31:16 | FRMEN | FRMSYNC | FRMPOL | MSSEN | 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 |
| 7610 | SPI4STAT | 31:16 | — | — | — | RXBUFELM<4:0> | | | | | — | — | — | TXBUFELM<4:0> | | | | | 0000 |
| | | 15:0 | — | — | — | FRMERR | SPIBUSY | — | — | SPITUR | SRMT | SPIROV | SPIRBE | — | SPITBE | — | SPITBF | SPIRBF | 0028 |
| 7620 | SPI4BUF | 31:16 | DATA<31:0> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | | | | | | | | | | | | | | | | | 0000 |
| 7630 | SPI4BRG | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | — | — | — | BRG<12:0> | | | | | | | | | | | | | 0000 |
| 7640 | SPI4CON2 | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | SPI SGNEXT | — | — | FRM ERREN | SPI ROVEN | SPI TUREN | IGNROV | IGNTUR | AUDEN | — | — | — | AUD MONO | — | AUDMOD<1:0> | | 0C00 |
| 7800 | SPI5CON | 31:16 | FRMEN | FRMSYNC | FRMPOL | MSSEN | 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 |
| 7810 | SPI5STAT | 31:16 | — | — | — | RXBUFELM<4:0> | | | | | — | — | — | TXBUFELM<4:0> | | | | | 0000 |
| | | 15:0 | — | — | — | FRMERR | SPIBUSY | — | — | SPITUR | SRMT | SPIROV | SPIRBE | — | SPITBE | — | SPITBF | SPIRBF | 0028 |
| 7820 | SPI5BUF | 31:16 | DATA<31:0> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | | | | | | | | | | | | | | | | | 0000 |
| 7830 | SPI5BRG | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | — | — | — | BRG<12:0> | | | | | | | | | | | | | 0000 |
| 7840 | SPI5CON2 | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | SPI SGNEXT | — | — | FRM ERREN | SPI ROVEN | SPI TUREN | IGNROV | IGNTUR | AUDEN | — | — | — | AUD 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 **13.2 "CLR, SET, and INV Registers"** for more information.

## TABLE 25-2: 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(1) | EIRDY22(1) | EIRDY21(1) | EIRDY20(1) | 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(1) | EIRDY46(1) | EIRDY45(1) | — | — | — | EIRDY41(1) | EIRDY40(1) | EIRDY39(1) | EIRDY38(1) | EIRDY37(1) | EIRDY36(1) | EIRDY35(1) | EIRDY34(1) | EIRDY33(1) | — | 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:** This register is for internal ADC input sources (i.e., $V_{BAT}$, 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.

**REGISTER 25-3: 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(1,2,3,4) | 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> | | | | | |

| Legend: | 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
10 = REFCLK3
01 = FRC
00 = PBCLK5

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.

**REGISTER 25-4: ADCTRGMODE: ADC TRIGGERING MODE FOR DEDICATED ADC REGISTER**

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:** ADC2Synchronous 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 which alternate input is selected by SHxALT, for ADC0-ADC5 only, 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 must initialize the ADC0 module as if AN24 were actually AN0.

**REGISTER 25-18: 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 = Reserved
- 11101 = PWM Generator 6 Current-Limit (Motor Control Variants Only)
- 11100 = PWM Generator 5 Current-Limit (Motor Control Variants Only)
- 11011 = PWM Generator 4 Current-Limit (Motor Control Variants Only)
- 11010 = PWM Generator 3 Current-Limit (Motor Control Variants Only)
- 11001 = PWM Generator 2 Current-Limit (Motor Control Variants Only)
- 11000 = PWM Generator 1 Current-Limit (Motor Control Variants Only)
- 10111 = Reserved
- 10110 = Reserved
- 10101 = Reserved
- 10100 = CTMU trip
- 10011 = Output Compare 4 (Rising Edge Only)
- 10010 = Output Compare 3 (Rising Edge Only)
- 10001 = Output Compare 2 (Rising Edge Only)
- 10000 = Output Compare 1 (Rising Edge Only)
- 01111 = PWM Generator 6 trigger (Motor Control Variants Only)
- 01110 = PWM Generator 5 trigger (Motor Control Variants Only)
- 01101 = PWM Generator 4 trigger (Motor Control Variants Only)
- 01100 = PWM Generator 3 trigger (Motor Control Variants Only)
- 01011 = PWM Generator 2 trigger (Motor Control Variants Only)
- 01010 = PWM Generator 1 trigger (Motor Control Variants Only)
- 01001 = Secondary Special Event trigger (Motor Control Variants Only)
- 01000 = Primary Special Event trigger (Motor Control Variants Only)
- 00111 = General Purpose Timer5
- 00110 = General Purpose Timer3
- 00101 = General Purpose Timer1
- 00100 = INT0
- 00011 = Scan trigger (see **Note**)
- 00010 = Software level trigger
- 00001 = Software edge trigger
- 00000 = No Trigger

**Note:** 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.

bit 23-21 **Unimplemented:** Read as '0'

**REGISTER 26-8: CxTMR: CAN TIMER REGISTER ('x' = 1-4)**

bit 15-0 **CANTSPRE<15:0>:** CAN Time Stamp Timer Prescaler bits

1111 1111 1111 1111 = CAN time stamp timer (CANTS) increments every 65,535 system clocks

.

.

.

0000 0000 0000 0000 = CAN time stamp timer (CANTS) increments every system clock

**Note 1:** CxTMR will be paused when CANCAP = 0.

**2:** The CxTMR prescaler count will be reset on any write to CxTMR (CANTSPRE will be unaffected).

**REGISTER 26-12: CxFLTCON2: CAN FILTER CONTROL REGISTER 2 ('x' = 1-4) (CONTINUED)**

bit 15 **FLTEN9:** Filter 9 Enable bit
1 = Filter is enabled
0 = Filter is disabled

bit 14-13 **MSEL9<1:0>:** Filter 9 Mask Select bits
11 = Reserved
10 = Acceptance Mask 2 is selected
01 = Acceptance Mask 1 is selected
00 = Acceptance Mask 0 is selected

bit 12-8 **FSEL9<4:0>:** FIFO Selection bits
11111 = Message matching filter is stored in FIFO buffer 31
11110 = Message matching filter is stored in FIFO buffer 30
.
.
.
00001 = Message matching filter is stored in FIFO buffer 1
00000 = Message matching filter is stored in FIFO buffer 0

bit 7 **FLTEN8:** Filter 8 Enable bit
1 = Filter is enabled
0 = Filter is disabled

bit 6-5 **MSEL8<1:0>:** Filter 8 Mask Select bits
11 = Reserved
10 = Acceptance Mask 2 is selected
01 = Acceptance Mask 1 is selected
00 = Acceptance Mask 0 is selected

bit 4-0 **FSEL8<4:0>:** FIFO Selection bits
11111 = Message matching filter is stored in FIFO buffer 31
11110 = Message matching filter is stored in FIFO buffer 30
.
.
.
00001 = Message matching filter is stored in FIFO buffer 1
00000 = Message matching filter is stored in FIFO buffer 0

**Note:** The bits in this register can only be modified if the corresponding filter enable (FLTENn) bit is '0'.

**NOTES:**

**TABLE 30-1: QEI1 THROUGH QEI6 REGISTER MAP (CONTINUED)**

| 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 | |
| B880 | INT4HLD | 31:16 | INTHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INTHLD<15:0> | | | | | | | | | | | | | | | | 0000 |
| B890 | INDX4CNT | 31:16 | INDXCNT<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INDXCNT<15:0> | | | | | | | | | | | | | | | | 0000 |
| B8A0 | INDX4HLD | 31:16 | INDXHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INDXHLD<15:0> | | | | | | | | | | | | | | | | 0000 |
| B8B0 | QEI4ICC | 31:16 | QEIICC<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | QEIICC<15:0> | | | | | | | | | | | | | | | | 0000 |
| B8C0 | QEI4CMPL | 31:16 | QEICMPL<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | QEICMPL<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA00 | QEI5CON | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | QEIEN | — | QEISIDL | PIMOD<2:0> | | | IMV<1:0> | | — | INTDIV<2:0> | | | CNTPOL | GATEN | CCM<1:0> | | 0000 |
| BA10 | QEI5IOC | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | HCAPEN | 0000 |
| | | 15:0 | QCAPEN | FLTREN | QFDIV<2:0> | | | OUTFNC<1:0> | | SWPAB | HOMPOL | IDXPOL | QEBPOL | QEAPOL | HOME | INDEX | QEB | QEA | 0000 |
| BA20 | QEI5STAT | 31:16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0000 |
| | | 15:0 | — | — | PCHEQIRQ | PCHEQIEN | PCLEQIRQ | PCLEQIEN | POSOVIRQ | POSOVIEN | PCIIRQ | PCIIEN | VELOVIRQ | VELOVIEN | HOMIRQ | HOMIEN | IDXIRQ | IDXIEN | 0000 |
| BA30 | POS5CNT | 31:16 | POSCNT<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | POSCNT<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA40 | POS5HLD | 31:16 | POSHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | POSHLD<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA50 | VEL5CNT | 31:16 | VELCNT<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | VELCNT<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA60 | VEL5HLD | 31:16 | VELHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | VELHLD<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA70 | INT5TMR | 31:16 | INTTMR<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INTTMR<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA80 | INT5HLD | 31:16 | INTHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INTHLD<15:0> | | | | | | | | | | | | | | | | 0000 |
| BA90 | INDX5CNT | 31:16 | INDXCNT<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INDXCNT<15:0> | | | | | | | | | | | | | | | | 0000 |
| BAA0 | INDX5HLD | 31:16 | INDXHLD<31:16> | | | | | | | | | | | | | | | | 0000 |
| | | 15:0 | INDXHLD<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 **Section 13.2 "CLR, SET, and INV Registers"** for more information.

**REGISTER 30-2: QEIxIOC: QEIx I/O 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 | R/W-0 |
| | — | — | — | — | — | — | — | HCAPEN |
| 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 |
| | QCAPEN | FLTREN | QFDIV<2:0> | | | OUTFNC<1:0> | | SWPAB |
| 7:0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R-x | R-x | R-x | R-x |
| | HOMPOL | IDXPOL | QEBPOL | QEAPOL | HOME | INDEX | QEB | QEA |

| **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 31-17 **Unimplemented:** Read as '0'

bit 16 **HCAPEN:** Position Counter Input Capture by Home Event Enable bit
1 = HOMEx input event (positive edge) triggers a position capture event
0 = HOMEx input event (positive edge) does not trigger a position capture event

bit 15 **QCAPEN:** Position Counter Input Capture Enable bit
1 = Positive edge detect of Home input triggers position capture function
0 = Home input event (positive edge) does not trigger a capture even

bit 14 **FLTREN:** QEA/QEB/INDX/HOMEx Digital Filter Enable bit
1 = Input Pin Digital filter is enabled
0 = Input Pin Digital filter is disabled (bypassed)

bit 13-11 **QFDIV<2:0>:** QEA/QEB/INDX/HOMEx Digital Input Filter Clock Select bits
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 10-9 **OUTFNC<1:0>:** QEI Module Output Function Mode Select bits
11 = The CNTCMPx pin goes high when POSxCNT ≤ QEIxCMPL or POSxCNT ≥ QEIxICCH
10 = The CNTCMPx pin goes high when POSxCNT ≤ QEIxCMPL
01 = The CNTCMPx pin goes high when POSxCNT ≥ QEIxICCH
00 = Output is disabled

bit 8 **SWPAB:** Swap QEA and QEB Inputs bit
1 = QEAx and QEBx are swapped prior to quadrature decoder logic
0 = QEAx and QEBx are not swapped

bit 7 **HOMPOL:** HOMEx Input Polarity Select bit
1 = Input is inverted
0 = Input is not inverted

bit 6 **IDXPOL:** INDXx Input Polarity Select bit
1 = Input is inverted
0 = Input is not inverted

bit 5 **QEBPOL:** QEBx Input Polarity Select bit
1 = Input is inverted
0 = Input is not inverted

## 31.0 MOTOR CONTROL PWM MODULE

**Note:** This data sheet summarizes the features of the PIC32MK GP/MC 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).

The PIC32MK GP/MC Family of devices support a dedicated Motor Control Pulse-Width Modulation (PWM) module with up to 12 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
- Eight PWM generators with complimentary output pairs
- Four additional PWM generators with single ended outputs
- 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
- Nine 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
- Redundant 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

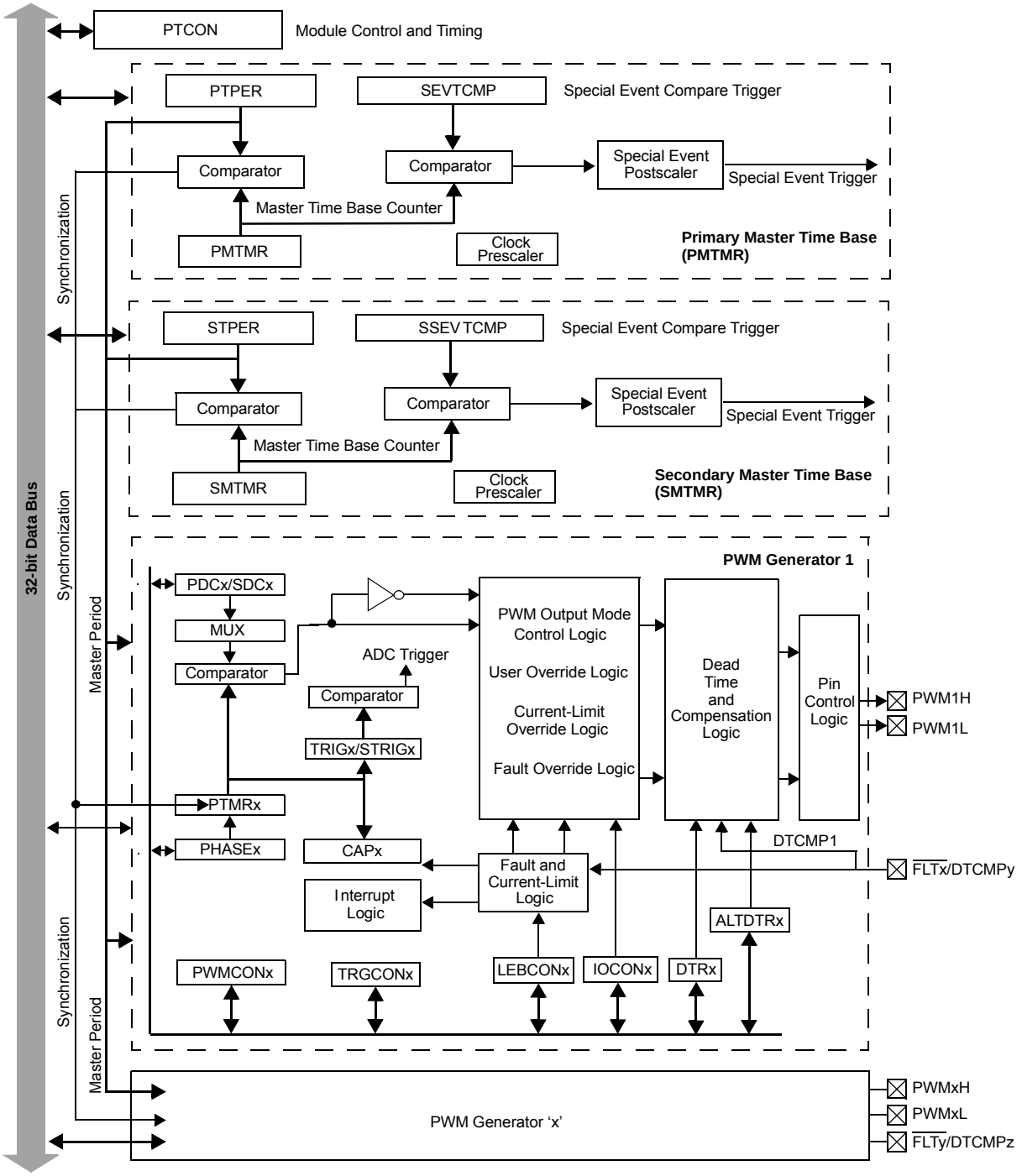
The Motor Control PWM module contains up to twelve PWM generators. Two master time base generators provide a synchronous signal as a common time base to synchronize the various PWM outputs. Each generator can operate independently or in synchronization with either of the two master time bases. The individual 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 6, 11 and 12 have two outputs, PWMxH and PWMxL, brought out to the dedicated pins. The PWM generators 7 through 10 have only the PWMxH outputs on pins, but can alternately be mapped onto PWMxL, where 'x' = 1-4, based on the PWMAPINx bit in the CFGCON register. Generators 11 and 12 have their PWMxH additionally brought out on the PWMxL pins of the generators 5 and 6, based on the PWMAPINx bit in the CFGCON register. The configuration bits PWMAPINx (CFGCON<23:18>) contain bits that help arbitrate which PWM output takes control of the I/O pin. This is in addition to PENx control bits which decide the if the MCPWM module of the I/O module assumes ownership of the output pin.

Figure 31-1 illustrates an architectural overview of the Motor Control PWM module and its interconnection with the CPU and other peripherals.

**FIGURE 31-2: MOTOR CONTROL PWM MODULE REGISTER INTERCONNECTION DIAGRAM**




**Legend:**

'x' = 1 through 12.
'y' = 1 through 8 and 15.
'z' = 3 through 8.

**Note:** Since DTCMPx and $\overline{\text{FLTx}}$ share the same digital input pins, their availability is mutually exclusive.

**REGISTER 31-15: PHASEx: PWM PRIMARY PHASE SHIFT REGISTER 'x' ('x' = 1 THROUGH 12)**

| 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 |
| | 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 | 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 **PHASE<15:0>:** PWM Phase Shift Value or Independent Time Base Period bits for the PWM Generator bits(6)

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:** PHASEx = (FSYSCLK / (FPWM * PCLKDIV<2:0> bits (PTCON<6:4>))
FPWM = User-desired PWM Frequency.

**REGISTER 31-20: 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)(1)

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(1, 2)

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(1)

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.

## 36.2 AC Characteristics and Timing Parameters

The information contained in this section defines PIC32MK GP/MC device AC characteristics and timing parameters.

**FIGURE 36-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**

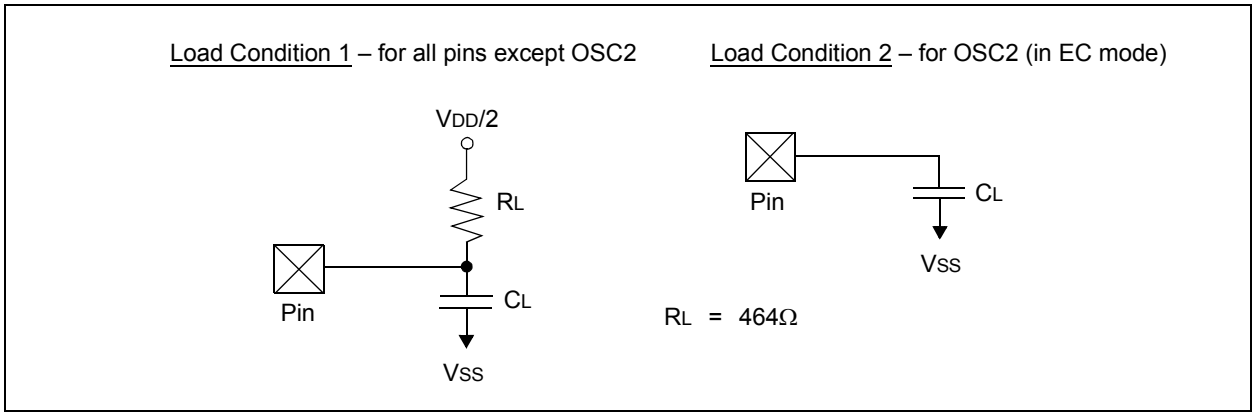



**TABLE 36-14: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS**

| AC CHARACTERISTICS | | | Standard Operating Conditions: 2.2V 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.(1)** | **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.