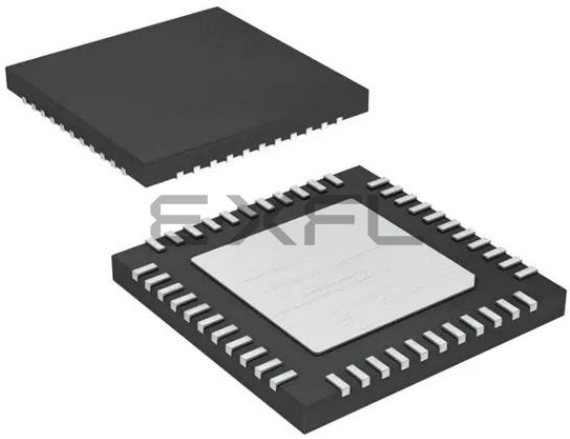
# Microchip Technology - DSPIC33EP512GM604-H/ML Datasheet



**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 | dsPIC |
| Core Size | 16-Bit |
| Speed | - |
| Connectivity | CANbus, I²C, IrDA, LINbus, QEI, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, DMA, I²S, Motor Control PWM, POR, PWM, WDT |
| Number of I/O | 35 |
| Program Memory Size | 512KB (170K x 24) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 48K x 8 |
| Voltage - Supply (Vcc/Vdd) | - |
| Data Converters | A/D 18x10b/12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 150°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 44-VQFN Exposed Pad |
| Supplier Device Package | 44-QFN (8x8) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/dspic33ep512gm604-h-ml |

## 3.7 Arithmetic Logic Unit (ALU)

The dsPIC33EPXXXGM3XX/6XX/7XX family ALU is 16 bits wide and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. Depending on the operation, the ALU can affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

The ALU can perform 8-bit or 16-bit operations, depending on the mode of the instruction that is used. Data for the ALU operation can come from the W register array or data memory, depending on the addressing mode of the instruction. Likewise, output data from the ALU can be written to the W register array or a data memory location.

Refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157) for information on the SR bits affected by each instruction.

The core CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

### 3.7.1 MULTIPLIER

Using the high-speed, 17-bit x 17-bit multiplier, the ALU supports unsigned, signed, or mixed-sign operation in several MCU multiplication modes:

- 16-bit x 16-bit signed
- 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 16-bit signed x 16-bit unsigned
- 16-bit unsigned x 5-bit (literal) unsigned
- 16-bit unsigned x 16-bit signed
- 8-bit unsigned x 8-bit unsigned

### 3.7.2 DIVIDER

The divide block supports 32-bit/16-bit and 16-bit/16-bit signed and unsigned integer divide operations with the following data sizes:

- 32-bit signed/16-bit signed divide
- 32-bit unsigned/16-bit unsigned divide
- 16-bit signed/16-bit signed divide
- 16-bit unsigned/16-bit unsigned divide

The quotient for all divide instructions ends up in W0 and the remainder in W1. 16-bit signed and unsigned `DIV` instructions can specify any W register for both the 16-bit divisor (Wn) and any W register (aligned) pair (W(m + 1):Wm) for the 32-bit dividend. The divide algorithm takes one cycle per bit of divisor, so both 32-bit/16-bit and 16-bit/16-bit instructions take the same number of cycles to execute.

## 3.8 DSP Engine

The DSP engine consists of a high-speed, 17-bit x 17-bit multiplier, a 40-bit barrel shifter and a 40-bit adder/subtracter (with two target accumulators, round and saturation logic).

The DSP engine can also perform inherent accumulator-to-accumulator operations that require no additional data. These instructions are `ADD`, `SUB` and `NEG`.

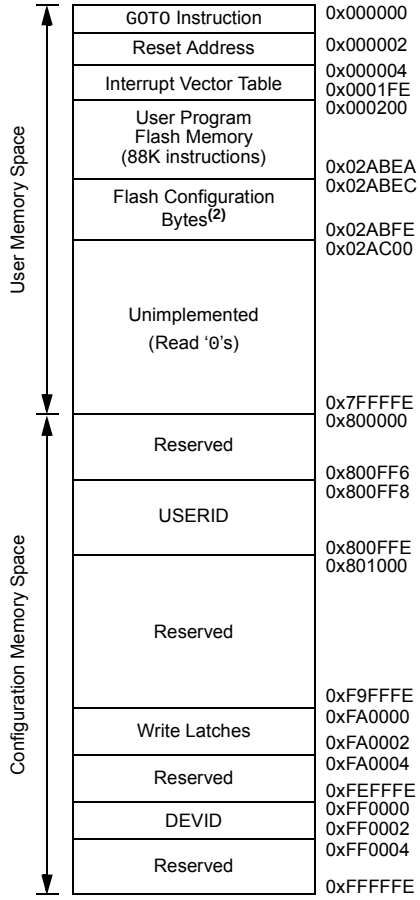
The DSP engine has options selected through bits in the CPU Core Control register (CORCON), as listed below:

- Fractional or integer DSP multiply (IF)
- Signed, unsigned or mixed-sign DSP multiply (US)
- Conventional or convergent rounding (RND)
- Automatic saturation on/off for ACCA (SATA)
- Automatic saturation on/off for ACCB (SATB)
- Automatic saturation on/off for writes to data memory (SATDW)
- Accumulator Saturation mode selection (ACCSAT)

**TABLE 3-2: DSP INSTRUCTIONS SUMMARY**

| Instruction | Algebraic Operation | ACC Write Back |
|---|---|---|
| `CLR` | $A = 0$ | Yes |
| `ED` | $A = (x - y)^2$ | No |
| `EDAC` | $A = A + (x - y)^2$ | No |
| `MAC` | $A = A + (x \bullet y)$ | Yes |
| `MAC` | $A = A + x^2$ | No |
| `MOVSAC` | No change in A | Yes |
| `MPY` | $A = x \bullet y$ | No |
| `MPY` | $A = x^2$ | No |
| `MPY.N` | $A = -x \bullet y$ | No |
| `MSC` | $A = A - x \bullet y$ | Yes |

**FIGURE 4-2: PROGRAM MEMORY MAP FOR dsPIC33EP256GM3XX/6XX/7XX DEVICES(1)**




**Note 1:** Memory areas are not shown to scale.

**2:** On Reset, these bits are automatically copied into the device Configuration Shadow registers.

**TABLE 4-28: CAN2 REGISTER MAP WHEN WIN (C1CTRL<0>) = 1 FOR dsPIC33EPXXXGM60X/7XX DEVICES(1)**

| SFR Name | Addr. | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | All Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0500-051E | See definition when WIN = x | | | | | | | | | | | | | | | | |
| C2BUFPNT1 | 0520 | F3BP3 | F3BP2 | F3BP1 | F3BP0 | F2BP3 | F2BP2 | F2BP1 | F2BP0 | F1BP3 | F1BP2 | F1BP1 | F1BP0 | F0BP3 | F0BP2 | F0BP1 | F0BP0 | 0000 |
| C2BUFPNT2 | 0522 | F7BP3 | F7BP2 | F7BP1 | F7BP0 | F6BP3 | F6BP2 | F6BP1 | F6BP0 | F5BP3 | F5BP2 | F5BP1 | F5BP0 | F4BP3 | F4BP2 | F4BP1 | F4BP0 | 0000 |
| C2BUFPNT3 | 0524 | F11BP3 | F11BP2 | F11BP1 | F11BP0 | F10BP3 | F10BP2 | F10BP1 | F10BP0 | F9BP3 | F9BP2 | F9BP1 | F9BP0 | F8BP3 | F8BP2 | F8BP1 | F8BP0 | 0000 |
| C2BUFPNT4 | 0526 | F15BP3 | F15BP2 | F15BP1 | F15BP0 | F14BP3 | F14BP2 | F14BP1 | F14BP0 | F13BP3 | F13BP2 | F13BP1 | F13BP0 | F12BP3 | F12BP2 | F12BP1 | F12BP0 | 0000 |
| C2RXM0SID | 0530 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXM0EID | 0532 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXM1SID | 0534 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXM1EID | 0536 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXM2SID | 0538 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXM2EID | 053A | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF0SID | 0540 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF0EID | 0542 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF1SID | 0544 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF1EID | 0546 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF2SID | 0548 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF2EID | 054A | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF3SID | 054C | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF3EID | 054E | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF4SID | 0550 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF4EID | 0552 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF5SID | 0554 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF5EID | 0556 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF6SID | 0558 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF6EID | 055A | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF7SID | 055C | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF7EID | 055E | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF8SID | 0560 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF8EID | 0562 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF9SID | 0564 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF9EID | 0566 | EID<15:0> | | | | | | | | | | | | | | | | xxxx |
| C2RXF10SID | 0568 | SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 | SID2 | SID1 | SID0 | — | MIDE | — | EID17 | EID16 | xxxx |
| C2RXF10EID | 056A | EID<15:0> | | | | | | | | | | | | | | | | xxxx |

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

**Note 1:** These registers are not present on dsPIC33EPXXXGM3XX devices.

**TABLE 4-42: CTMU REGISTER MAP**

| SFR Name | Addr. | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | All Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CTMUCON1 | 033A | CTMUEN | — | CTMUSIDL | TGEN | EDGEN | EDGSEQEN | IDISSEN | CTTRIG | — | — | — | — | — | — | — | — | 0000 |
| CTMUCON2 | 033C | EDG1MOD | EDG1POL | EDG1SEL3 | EDG1SEL2 | EDG1SEL1 | EDG1SEL0 | EDG2STAT | EDG1STAT | EDG2MOD | EDG2POL | EDG2SEL3 | EDG2SEL2 | EDG2SEL1 | EDG2SEL0 | — | — | 0000 |
| CTMUICON | 033E | ITRIM5 | ITRIM4 | ITRIM3 | ITRIM2 | ITRIM1 | ITRIM0 | IRNG1 | IRNG0 | — | — | — | — | — | — | — | — | 0000 |

**Legend:** — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

**TABLE 4-43: JTAG INTERFACE REGISTER MAP**

| SFR Name | Addr. | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | All Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| JDATAH | 0FF0 | — | — | — | — | JDATAH<27:16> | | | | | | | | | | | | xxxx |
| JDATAL | 0FF2 | JDATAL<15:0> | | | | | | | | | | | | | | | | 0000 |

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

**TABLE 4-44: REAL-TIME CLOCK AND CALENDAR REGISTER MAP**

| File Name | Addr. | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | All Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ALRMVAL | 0620 | Alarm Value Register Window Based on ALRMPTR<1:0> | | | | | | | | | | | | | | | | xxxx |
| ALCFGRPT | 0622 | ALRMEN | CHIME | AMASK3 | AMASK2 | AMASK1 | AMASK0 | ALRMPTR1 | ALRMPTR0 | ARPT7 | ARPT6 | ARPT5 | ARPT4 | ARPT3 | ARPT2 | ARPT1 | ARPT0 | 0000 |
| RTCVAL | 0624 | RTCC Value Register Window Based on RTCPTR<1:0> | | | | | | | | | | | | | | | | xxxx |
| RCFGCAL | 0626 | RTCEN | — | RTCWREN | RTCSYNC | HALFSEC | RTCOE | RTCPTR1 | RTCPTR0 | CAL7 | CAL6 | CAL5 | CAL4 | CAL3 | CAL2 | CAL1 | CAL0 | 0000 |

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

## 4.7 Interfacing Program and Data Memory Spaces

The dsPIC33EPXXXGM3XX/6XX/7XX architecture uses a 24-bit-wide Program Space and a 16-bit-wide Data Space. The architecture is also a modified Harvard scheme, meaning that data can also be present in the Program Space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Aside from normal execution, the architecture of the dsPIC33EPXXXGM3XX/6XX/7XX devices provides two methods by which Program Space can be accessed during operation:

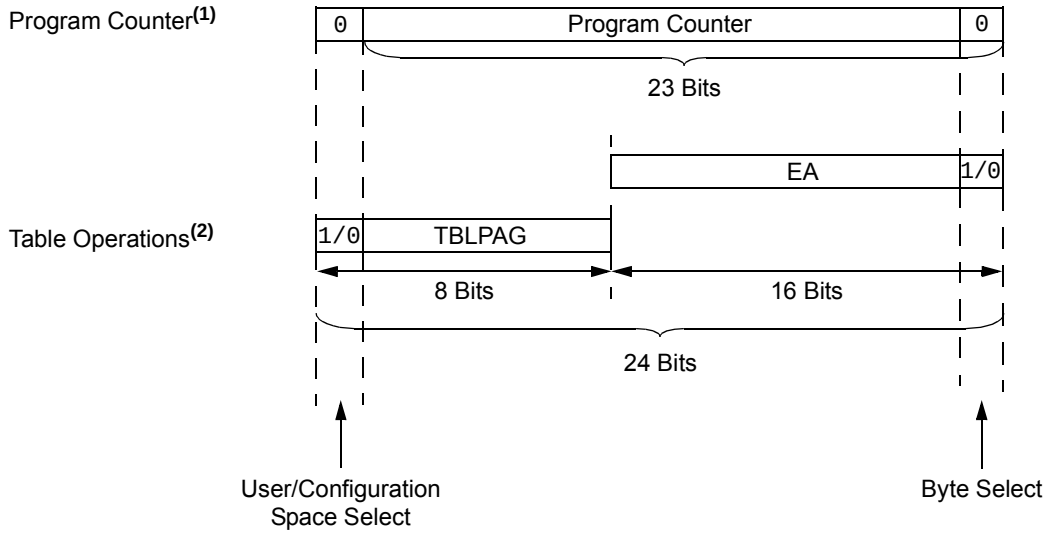
- Using table instructions to access individual bytes or words anywhere in the Program Space
- Remapping a portion of the Program Space into the Data Space (Program Space Visibility)

Table instructions allow an application to read or write to small areas of the program memory. This capability makes the method ideal for accessing data tables that need to be updated periodically. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look-ups from a large table of static data. The application can only access the least significant word of the program word.

**TABLE 4-68: PROGRAM SPACE ADDRESS CONSTRUCTION**

| Access Type | Access Space | Program Space Address | | | | |
|---|---|---|---|---|---|---|
| | | <23> | <22:16> | <15> | <14:1> | <0> |
| Instruction Access (Code Execution) | User | 0 | PC<22:1> | | | 0 |
| | | 0xx xxxx xxxx xxxx xxxx xxx0 | | | | |
| TBLRD/TBLWT (Byte/Word Read/Write) | User | TBLPAG<7:0> | | Data EA<15:0> | | |
| | | 0xxx xxxx | | xxxx xxxx xxxx xxxx | | |
| | Configuration | TBLPAG<7:0> | | Data EA<15:0> | | |
| | | 1xxx xxxx | | xxxx xxxx xxxx xxxx | | |

**FIGURE 4-16: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION**




**Note 1:** The Least Significant bit (LSb) of Program Space addresses is always fixed as '0' to maintain word alignment of data in the Program and Data Spaces.

**2:** Table operations are not required to be word-aligned. Table Read operations are permitted in the configuration memory space.

**TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)**

| Interrupt Source | Vector # | IRQ # | IVT Address | Interrupt Bit Location | | |
|---|---|---|---|---|---|---|
| | | | | Flag | Enable | Priority |
| OC5 – Output Compare 5 | 49 | 41 | 0x000066 | IFS2<9> | IEC2<9> | IPC10<6:4> |
| OC6 – Output Compare 6 | 50 | 42 | 0x000068 | IFS2<10> | IEC2<10> | IPC10<10:8> |
| OC7 – Output Compare 7 | 51 | 43 | 0x00006A | IFS2<11> | IEC2<11> | IPC10<14:12> |
| OC8 – Output Compare 8 | 52 | 44 | 0x00006C | IFS2<12> | IEC2<12> | IPC11<2:0> |
| PMP – Parallel Master Port(2) | 53 | 45 | 0x00006E | IFS2<13> | IEC2<13> | IPC11<6:4> |
| Reserved | 54 | 46 | 0x000070 | — | — | — |
| T6 – Timer6 | 55 | 47 | 0x000072 | IFS2<15> | IEC2<15> | IPC11<14:12> |
| T7 – Timer7 | 56 | 48 | 0x000074 | IFS3<0> | IEC3<0> | IPC12<2:0> |
| SI2C2 – I2C2 Slave Event | 57 | 49 | 0x000076 | IFS3<1> | IEC3<1> | IPC12<6:4> |
| MI2C2 – I2C2 Master Event | 58 | 50 | 0x000078 | IFS3<2> | IEC3<2> | IPC12<10:8> |
| T8 – Timer8 | 59 | 51 | 0x00007A | IFS3<3> | IEC3<3> | IPC12<14:12> |
| T9 – Timer9 | 60 | 52 | 0x00007C | IFS3<4> | IEC3<4> | IPC13<2:0> |
| INT3 – External Interrupt 3 | 61 | 53 | 0x00007E | IFS3<5> | IEC3<5> | IPC13<6:4> |
| INT4 – External Interrupt 4 | 62 | 54 | 0x000080 | IFS3<6> | IEC3<6> | IPC13<10:8> |
| C2RX – CAN2 RX Data Ready(1) | 63 | 55 | 0x000082 | IFS3<7> | IEC3<7> | IPC13<14:12> |
| C2 – CAN2 Event(1) | 64 | 56 | 0x000084 | IFS3<8> | IEC3<8> | IPC14<2:0> |
| PSEM – PCPWM Primary Event | 65 | 57 | 0x000086 | IFS3<9> | IEC3<9> | IPC14<6:4> |
| QEI1 – QEI1 Position Counter Compare | 66 | 58 | 0x000088 | IFS3<10> | IEC3<10> | IPC14<10:8> |
| DCIE – DCI Fault Interrupt | 67 | 59 | 0x00008A | IFS3<11> | IEC3<11> | IPC14<14:12> |
| DCI – DCI Transfer Done | 68 | 60 | 0x00008C | IFS3<12> | IEC3<12> | IPC15<2:0> |
| Reserved | 69 | 61 | 0x00008E | — | — | — |
| RTCC – Real-Time Clock and Calendar(2) | 70 | 62 | 0x000090 | IFS3<14> | IEC3<14> | IPC15<10:8> |
| Reserved | 71-72 | 63-64 | 0x000092-0x000094 | — | — | — |
| U1E – UART1 Error Interrupt | 73 | 65 | 0x000096 | IFS4<1> | IEC4<1> | IPC16<6:4> |
| U2E – UART2 Error Interrupt | 74 | 66 | 0x000098 | IFS4<2> | IEC4<2> | IPC16<10:8> |
| CRC – CRC Generator Interrupt | 75 | 67 | 0x00009A | IFS4<3> | IEC4<3> | IPC16<14:12> |
| Reserved | 76-77 | 68-69 | 0x00009C-0x00009E | — | — | — |
| C1TX – CAN1 TX Data Request(1) | 78 | 70 | 0x0000A0 | IFS4<6> | IEC4<6> | IPC17<10:8> |
| C2TX – CAN2 TX Data Request (1) | 79 | 71 | 0x0000A2 | IFS4<7> | IEC4<7> | IPC17<14:12> |
| Reserved | 80 | 72 | 0x0000A4 | — | — | — |
| PSESM – PCPWM Secondary Event | 81 | 73 | 0x0000A6 | IFS4<9> | IEC4<9> | IPC18<6:4> |
| Reserved | 82 | 74 | 0x0000A8 | — | — | — |
| QEI2 – QEI2 Position Counter Compare | 83 | 75 | 0x0000AA | IFS4<11> | IEC4<11> | IPC18<14:12> |
| Reserved | 84 | 76 | 0x0000AC | — | — | — |
| CTMU – CTMU Interrupt | 85 | 77 | 0x0000AE | IFS4<13> | IEC4<13> | IPC19<6:4> |
| Reserved | 86-88 | 78-80 | 0x0000B0-0x0000B4 | — | — | — |
| U3E – UART3 Error Interrupt | 89 | 81 | 0x0000B6 | IFS5<1> | IEC5<1> | IPC20<6:4> |
| U3RX – UART3 Receiver | 90 | 82 | 0x0000B8 | IFS5<2> | IEC5<2> | IPC20<10:8> |
| U3TX – UART3 Transmitter | 91 | 83 | 0x0000BA | IFS5<3> | IEC5<3> | IPC20<14:12> |
| Reserved | 92-94 | 84-86 | 0x0000BC-0x0000C0 | — | — | — |
| U4E – UART4 Error Interrupt | 95 | 87 | 0x0000C2 | IFS5<7> | IEC5<7> | IPC21<14:12> |
| U4RX – UART4 Receiver | 96 | 88 | 0x0000C4 | IFS5<8> | IEC5<8> | IPC22<2:0> |
| U4TX – UART4 Transmitter | 97 | 89 | 0x0000C6 | IFS5<9> | IEC5<9> | IPC22<6:4> |

**Note 1:** This interrupt source is available on dsPIC33EPXXXGM6XX/7XX devices only.

**2:** This interrupt source is not available on 44-pin devices.

**TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)**

| Interrupt Source | Vector # | IRQ # | IVT Address | Interrupt Bit Location | | |
|---|---|---|---|---|---|---|
| | | | | Flag | Enable | Priority |
| SPI3E – SPI3 Error | 98 | 90 | 0x0000C8 | IFS5<10> | IEC5<10> | IPC22<10:8> |
| SPI3 – SPI3 Transfer Done | 99 | 91 | 0x0000CA | IFS5<11> | IEC5<11> | IPC22<14:12> |
| Reserved | 100-101 | 92-93 | 0x0000CC-0x0000CE | — | — | — |
| PWM1 – PWM Generator 1 | 102 | 94 | 0x0000D0 | IFS5<14> | IEC5<14> | IPC23<10:8> |
| PWM2 – PWM Generator 2 | 103 | 95 | 0x0000D2 | IFS5<15> | IEC5<15> | IPC23<14:12> |
| PWM3 – PWM Generator 3 | 104 | 96 | 0x0000D4 | IFS6<0> | IEC6<0> | IPC24<2:0> |
| PWM4 – PWM Generator 4 | 105 | 97 | 0x0000D6 | IFS6<1> | IEC6<1> | IPC24<6:4> |
| PWM5 – PWM Generator 5 | 106 | 98 | 0x0000D8 | IFS6<2> | IEC6<2> | IPC24<10:8> |
| PWM6 – PWM Generator 6 | 107 | 99 | 0x0000DA | IFS6<3> | IEC6<3> | IPC24<14:12> |
| Reserved | 108-149 | 100-141 | 0x0000DC-0x00012E | — | — | — |
| ICD – ICD Application | 150 | 142 | 0x000142 | IFS8<14> | IEC8<14> | IPC35<10:8> |
| JTAG – JTAG Programming | 151 | 143 | 0x000130 | IFS8<15> | IEC8<15> | IPC35<14:12> |
| Reserved | 152 | 144 | 0x000134 | — | — | — |
| PTGSTEP – PTG Step | 153 | 145 | 0x000136 | IFS9<1> | IEC9<1> | IPC36<6:4> |
| PTGWDT – PTG Watchdog Time-out | 154 | 146 | 0x000138 | IFS9<2> | IEC9<2> | IPC36<10:8> |
| PTG0 – PTG Interrupt 0 | 155 | 147 | 0x00013A | IFS9<3> | IEC9<3> | IPC36<14:12> |
| PTG1 – PTG Interrupt 1 | 156 | 148 | 0x00013C | IFS9<4> | IEC9<4> | IPC37<2:0> |
| PTG2 – PTG Interrupt 2 | 157 | 149 | 0x00013E | IFS9<5> | IEC9<5> | IPC37<6:4> |
| PTG3 – PTG Interrupt 3 | 158 | 150 | 0x000140 | IFS9<6> | IEC9<6> | IPC37<10:8> |
| Reserved | 159-245 | 151-245 | 0x000142-0x0001FE | — | — | — |
| Lowest Natural Order Priority | | | | | | |

**Note 1:** This interrupt source is available on dsPIC33EPXXXGM6XX/7XX devices only.

**2:** This interrupt source is not available on 44-pin devices.

**REGISTER 8-2: DMAxREQ: DMA CHANNEL x IRQ SELECT REGISTER**

| R/S-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| FORCE(1) | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| IRQSEL7 | IRQSEL6 | IRQSEL5 | IRQSEL4 | IRQSEL3 | IRQSEL2 | IRQSEL1 | IRQSEL0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | 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 15 **FORCE:** Force DMA Transfer bit(1)
1 = Forces a single DMA transfer (Manual mode)
0 = Automatic DMA transfer initiation by DMA request

bit 14-8 **Unimplemented:** Read as '0'

bit 7-0 **IRQSEL<7:0>:** DMA Peripheral IRQ Number Select bits
01011011 = SPI3 – Transfer done
01011001 = UART4TX – UART4 transmitter
01011000 = UART4RX – UART4 receiver
01010011 = UART3TX – UART3 transmitter
01010010 = UART3RX – UART3 receiver
01000111 = CAN2 – TX data request
01000110 = CAN1 – TX data request
00111100 = DCI – Codec transfer done
00110111 = CAN2 – RX data ready
00101101 = PMP – PMP data move
00100110 = IC4 – Input Capture 4
00100101 = IC3 – Input Capture 3
00100010 = CAN1 – RX data ready
00100001 = SPI2 – SPI2 transfer done
00011111 = UART2TX – UART2 transmitter
00011110 = UART2RX – UART2 receiver
00011100 = TMR5 – Timer5
00011011 = TMR4 – Timer4
00011010 = OC4 – Output Compare 4
00011001 = OC3 – Output Compare 3
00010101 = ADC2 – ADC2 convert done
00001101 = ADC1 – ADC1 convert done
00001100 = UART1TX – UART1 transmitter
00001011 = UART1RX – UART1 receiver
00001010 = SPI1 – SPI1 transfer done
00001000 = TMR3 – Timer3
00000111 = TMR2 – Timer2
00000110 = OC2 – Output Compare 2
00000101 = IC2 – Input Capture 2
00000010 = OC1 – Output Compare 1
00000001 = IC1 – Input Capture 1
00000000 = INT0 – External Interrupt 0

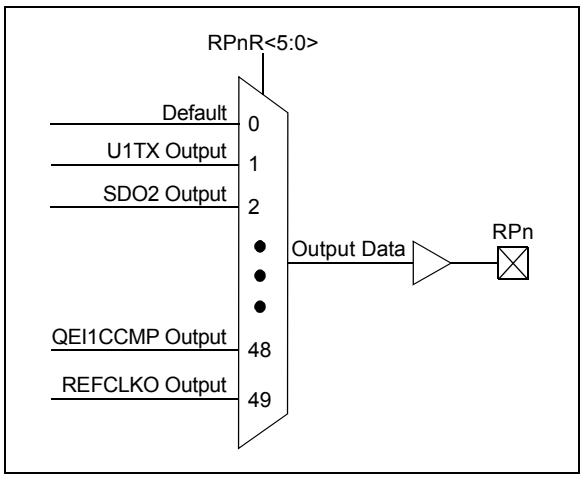
**Note 1:** The FORCE bit cannot be cleared by user software. The FORCE bit is cleared by hardware when the forced DMA transfer is complete or the channel is disabled (CHEN = 0).

### 11.4.5 OUTPUT MAPPING

In contrast to inputs, the outputs of the Peripheral Pin Select 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 RPORx registers are used to control output mapping. Like the RPINRx registers, each register contains sets of 6-bit fields, with each set associated with one RPn pin (see Register 11-30 through Register 11-42). The value of the bit field corresponds to one of the peripherals and that peripheral's output is mapped to the pin (see Table 11-3 and Figure 11-3).

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 11-3: MULTIPLEXING REMAPPABLE OUTPUT FOR RPn**




#### 11.4.5.1 Mapping Limitations

The control schema of the peripheral select pins is not limited to a small range of fixed peripheral configurations. There are no mutual or hardware-enforced lockouts between any of the peripheral mapping SFRs. Literally any combination of peripheral mappings across any or all of the RPn pins is possible. This includes both many-to-one and one-to-many mappings of peripheral inputs and outputs to pins. While such mappings may be technically possible from a configuration point of view, they may not be supportable from an electrical point of view.

**REGISTER 11-4: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7**

| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | IC2R<6:0> | | | | | | |
| bit 15 | | | | | | | bit 8 |

| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | IC1R<6:0> | | | | | | |
| bit 7 | | | | | | | bit 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 15 **Unimplemented:** Read as ‘0’

bit 14-8 **IC2R<6:0>:** Assign Input Capture 2 (IC2) to the Corresponding RPn Pin bits (see Table 11-2 for input pin selection numbers)

1111100 = Input tied to RPI124
•
•
•
0000001 = Input tied to CMP1
0000000 = Input tied to Vss

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **IC1R<6:0>:** Assign Input Capture 1 (IC1) to the Corresponding RPn Pin bits (see Table 11-2 for input pin selection numbers)

1111100 = Input tied to RPI124
•
•
•
0000001 = Input tied to CMP1
0000000 = Input tied to Vss

**REGISTER 16-7: STPER: PWMx SECONDARY MASTER TIME BASE PERIOD REGISTER**

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| STPER<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 8 |

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| STPER<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **STPER<15:0>:** PWMx Secondary Master Time Base (PMTMR) Period Value bits

**REGISTER 16-8: SSEVTCMP: PWMx SECONDARY SPECIAL EVENT COMPARE REGISTER**

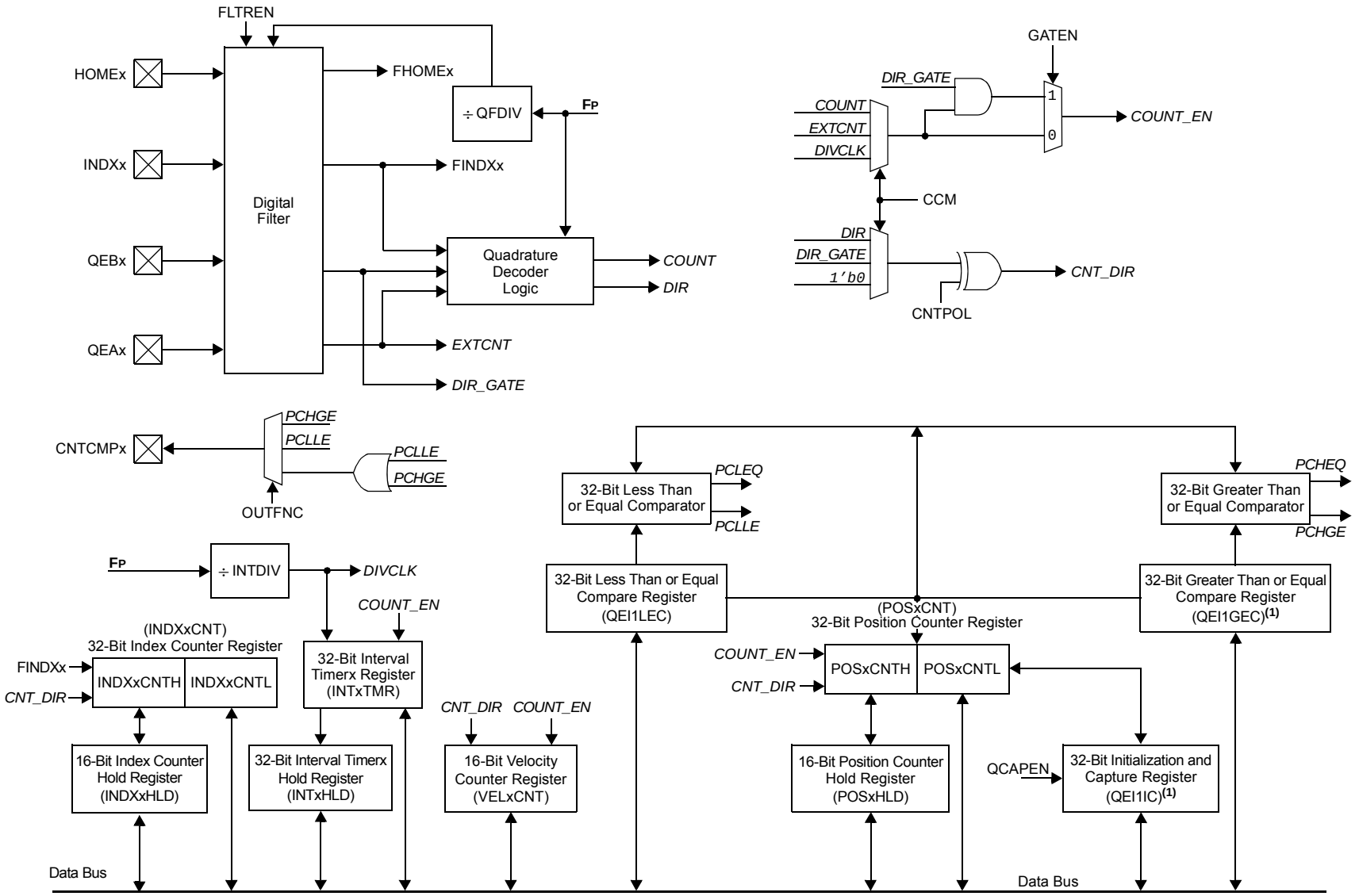
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| SSEVTCMP<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| SSEVTCMP<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **SSEVTCMP<15:0>:** PWMx Secondary Special Event Compare Count Value bits

**FIGURE 17-1: QEIx BLOCK DIAGRAM**




**Note 1:** These registers map to the same memory location.

**REGISTER 17-6: POSxHLD: POSITION COUNTER x HOLD REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| POSHLD<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| POSHLD<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **POSHLD<15:0>:** Holding Register for Reading and Writing POSxCNT bits

**REGISTER 17-7: VELxCNT: VELOCITY COUNTER x REGISTER**

| 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> | | | | | | | |
| bit 15 | | | | | | | bit 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<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **VELCNT<15:0>:** Velocity Counter x bits

**REGISTER 17-10: INDXxHLD: INDEX COUNTER x HOLD REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| INDXHLD<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| INDXHLD<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **INDXHLD<15:0>:** Holding Register for Reading and Writing INDXxCNT bits

**REGISTER 17-11: QEIxICH: QEIx INITIALIZATION/CAPTURE HIGH WORD REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| QEIIC<31:24> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| QEIIC<23:16> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **QEIIC<31:16>:** High Word Used to Form 32-Bit Initialization/Capture Register (QEIxIC) bits

**REGISTER 17-12: QEIxICL: QEIx INITIALIZATION/CAPTURE LOW WORD REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| QEIIC<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| QEIIC<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **QEIIC<15:0>:** Low Word Used to Form 32-Bit Initialization/Capture Register (QEIxIC) bits

## 18.1 SPI Helpful Tips

1. In Frame mode, if there is a possibility that the master may not be initialized before the slave:
   a) If FRMPOL (SPIxCON2<13>) = 1, use a pull-down resistor on $\overline{SSx}$.
   b) If FRMPOL = 0, use a pull-up resistor on $\overline{SSx}$.

| **Note:** | This insures that the first frame transmission after initialization is not shifted or corrupted. |
|---|---|

2. In Non-Framed 3-Wire mode (i.e., not using $\overline{SSx}$ from a master):
   a) If CKP (SPIxCON1<6>) = 1, always place a pull-up resistor on $\overline{SSx}$.
   b) If CKP = 0, always place a pull-down resistor on $\overline{SSx}$.

| **Note:** | This will insure that during power-up and initialization, the master/slave will not lose sync due to an errant SCK transition that would cause the slave to accumulate data shift errors, for both transmit and receive, appearing as corrupted data. |
|---|---|

3. FRMEN (SPIxCON2<15>) = 1 and SSEN (SPIxCON1<7>) = 1 are exclusive and invalid. In Frame mode, SCKx is continuous and the Frame Sync pulse is active on the $\overline{SSx}$ pin, which indicates the start of a data frame.

| **Note:** | Not all third-party devices support Frame mode timing. Refer to the SPIx specifications in **Section 33.0 "Electrical Characteristics"** for details. |
|---|---|

4. In Master mode only, set the SMP bit (SPIxCON1<9>) to a '1' for the fastest SPI data rate possible. The SMP bit can only be set at the same time or after the MSTEN bit (SPIxCON1<5>) is set.

To avoid invalid slave read data to the master, the user's master software must ensure enough time for slave software to fill its write buffer before the user application initiates a master write/read cycle. It is always advisable to preload the SPIxBUF Transmit register in advance of the next master transaction cycle. SPIxBUF is transferred to the SPIx Shift register and is empty once the data transmission begins.

**REGISTER 26-4: CMxMSKSRC: COMPARATOR x MASK SOURCE SELECT CONTROL REGISTER**

| U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | RW-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | SELSRCC3 | SELSRCC2 | SELSRCC1 | SELSRCC0 |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| SELSRCB3 | SELSRCB2 | SELSRCB1 | SELSRCB0 | SELSRCA3 | SELSRCA2 | SELSRCA1 | SELSRCA0 |
| bit 7 | | | | | | | bit 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 15-12 **Unimplemented:** Read as ‘0’

bit 11-8 **SELSRCC<3:0>:** Mask C Input Select bits

1111 = FLT4
1110 = FLT2
1101 = PTGO19
1100 = PTGO18
1011 = PWM6H
1010 = PWM6L
1001 = PWM5H
1000 = PWM5L
0111 = PWM4H
0110 = PWM4L
0101 = PWM3H
0100 = PWM3L
0011 = PWM2H
0010 = PWM2L
0001 = PWM1H
0000 = PWM1L

bit 7-4 **SELSRCB<3:0>:** Mask B Input Select bits

1111 = FLT4
1110 = FLT2
1101 = PTGO19
1100 = PTGO18
1011 = PWM6H
1010 = PWM6L
1001 = PWM5H
1000 = PWM5L
0111 = PWM4H
0110 = PWM4L
0101 = PWM3H
0100 = PWM3L
0011 = PWM2H
0010 = PWM2L
0001 = PWM1H
0000 = PWM1L

**REGISTER 29-3: CRCXORH: CRC XOR POLYNOMIAL HIGH REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| X<31:24> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| X<23:16> | | | | | | | |
| bit 7 | | | | | | | bit 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 15-0 **X<31:16>:** XOR of Polynomial Term $X^n$ Enable bits

**REGISTER 29-4: CRCXORL: CRC XOR POLYNOMIAL LOW REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| X<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 |
|---|---|---|---|---|---|---|---|
| X<7:1> | | | | | | | — |
| bit 7 | | | | | | | bit 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 15-1 **X<15:1>:** XOR of Polynomial Term $X^n$ Enable bits

bit 0 **Unimplemented:** Read as ‘0’

**TABLE 30-1: CONFIGURATION BYTE REGISTER MAP**

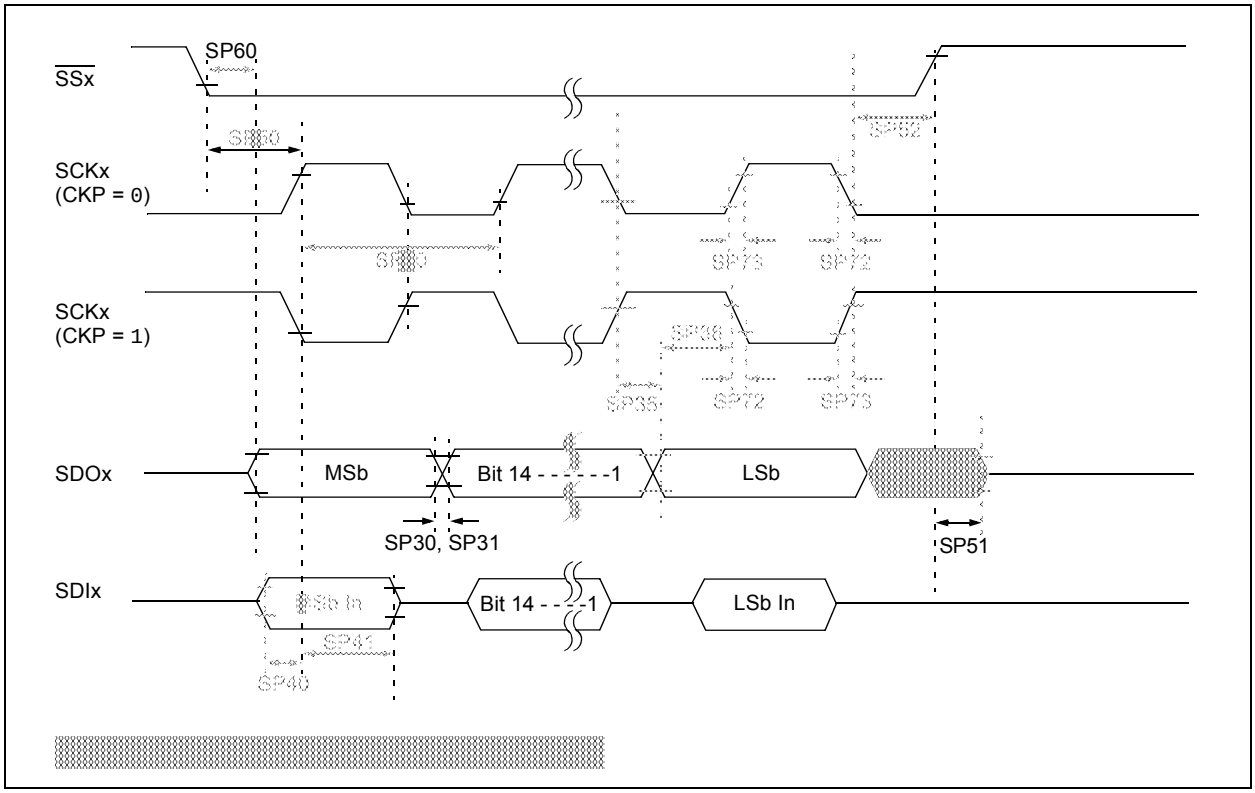
| File Name | Address | Device Memory Size (Kbytes) | Bits 23-8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Reserved | 0157EC | 128 | — | — | — | — | — | — | — | — | — |
| | 02AFEC | 256 | | | | | | | | | |
| | 0557EC | 512 | | | | | | | | | |
| Reserved | 0157EE | 128 | — | — | — | — | — | — | — | — | — |
| | 02AFEE | 256 | | | | | | | | | |
| | 0557EE | 512 | | | | | | | | | |
| FICD | 0157F0 | 128 | — | Reserved(2) | — | JTAGEN | Reserved(1) | Reserved(2) | — | ICS<1:0> | |
| | 02AFF0 | 256 | | | | | | | | | |
| | 0557F0 | 512 | | | | | | | | | |
| FPOR | 0157F2 | 128 | — | WDTWIN<1:0> | | ALTI2C2 | ALTI2C1 | BOREN | — | — | — |
| | 02AFF2 | 256 | | | | | | | | | |
| | 0557F2 | 512 | | | | | | | | | |
| FWDT | 0157F4 | 128 | — | FWDTEN | WINDIS | PLLKEN | WDTPRE | WDTPOST<3:0> | | | |
| | 02AFF4 | 256 | | | | | | | | | |
| | 0557F4 | 512 | | | | | | | | | |
| FOSC | 0157F6 | 128 | — | FCKSM<1:0> | | IOL1WAY | — | — | OSCIOFNC | POSCMD<1:0> | |
| | 02AFF6 | 256 | | | | | | | | | |
| | 0557F6 | 512 | | | | | | | | | |
| FOSCSEL | 0157F8 | 128 | — | IESO | PWMLOCK | — | — | — | FNOSC<2:0> | | |
| | 02AFF8 | 256 | | | | | | | | | |
| | 0557F8 | 512 | | | | | | | | | |
| FGS | 0157FA | 128 | — | — | — | — | — | — | — | GCP | GWRP |
| | 02AFFA | 256 | | | | | | | | | |
| | 0557FA | 512 | | | | | | | | | |
| Reserved | 0157FC | 128 | — | — | — | — | — | — | — | — | — |
| | 02AFFC | 256 | | | | | | | | | |
| | 0557FC | 512 | | | | | | | | | |
| Reserved | 0157FE | 128 | — | — | — | — | — | — | — | — | — |
| | 02AFFE | 256 | | | | | | | | | |
| | 0557FE | 512 | | | | | | | | | |

**Legend:** — = unimplemented, read as '1'.

**Note 1:** This bit is reserved and must be programmed as '0'.

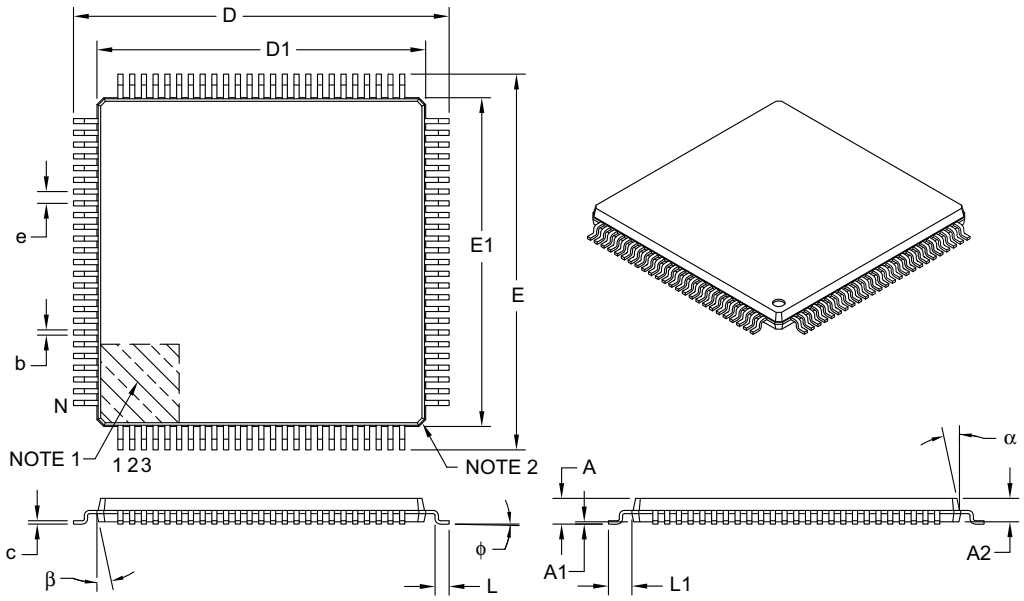
**2:** This bit is reserved and must be programmed as '1'.

**FIGURE 33-19: SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**



### 100-Lead Plastic Thin Quad Flatpack (PF) – 14x14x1 mm Body, 2.00 mm [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 100 | |
| 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 | ϕ | 0° | 3.5° | 7° |
| Overall Width | E | | 16.00 BSC | |
| Overall Length | D | | 16.00 BSC | |
| Molded Package Width | E1 | | 14.00 BSC | |
| Molded Package Length | D1 | | 14.00 BSC | |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.17 | 0.22 | 0.27 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. 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 C04-110B