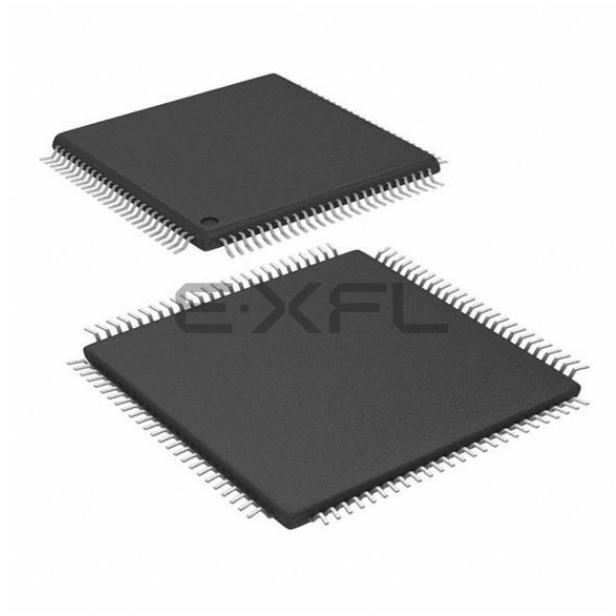
# Microchip Technology - DSPIC33FJ128GP710AT-I/PT Datasheet



**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 | 40 MIPs |
| Connectivity | CANbus, I²C, IrDA, LINbus, SPI, UART/USART |
| Peripherals | AC'97, Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT |
| Number of I/O | 85 |
| Program Memory Size | 128KB (128K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 16K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 3.6V |
| Data Converters | A/D 32x10b/12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 100-TQFP |
| Supplier Device Package | 100-TQFP (12x12) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj128gp710at-i-pt |

**TABLE 4-2: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXGPX10A DEVICES**

| SFR Name | SFR 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CNEN1 | 0060 | CN15IE | CN14IE | CN13IE | CN12IE | CN11IE | CN10IE | CN9IE | CN8IE | CN7IE | CN6IE | CN5IE | CN4IE | CN3IE | CN2IE | CN1IE | CN0IE | 0000 |
| CNEN2 | 0062 | — | — | — | — | — | — | — | — | CN23IE | CN22IE | CN21IE | CN20IE | CN19IE | CN18IE | CN17IE | CN16IE | 0000 |
| CNPU1 | 0068 | CN15PUE | CN14PUE | CN13PUE | CN12PUE | CN11PUE | CN10PUE | CN9PUE | CN8PUE | CN7PUE | CN6PUE | CN5PUE | CN4PUE | CN3PUE | CN2PUE | CN1PUE | CN0PUE | 0000 |
| CNPU2 | 006A | — | — | — | — | — | — | — | — | CN23PUE | CN22PUE | CN21PUE | CN20PUE | CN19PUE | CN18PUE | CN17PUE | CN16PUE | 0000 |

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

**TABLE 4-3: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXGPX08A DEVICES**

| SFR Name | SFR 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CNEN1 | 0060 | CN15IE | CN14IE | CN13IE | CN12IE | CN11IE | CN10IE | CN9IE | CN8IE | CN7IE | CN6IE | CN5IE | CN4IE | CN3IE | CN2IE | CN1IE | CN0IE | 0000 |
| CNEN2 | 0062 | — | — | — | — | — | — | — | — | — | — | CN21IE | CN20IE | CN19IE | CN18IE | CN17IE | CN16IE | 0000 |
| CNPU1 | 0068 | CN15PUE | CN14PUE | CN13PUE | CN12PUE | CN11PUE | CN10PUE | CN9PUE | CN8PUE | CN7PUE | CN6PUE | CN5PUE | CN4PUE | CN3PUE | CN2PUE | CN1PUE | CN0PUE | 0000 |
| CNPU2 | 006A | — | — | — | — | — | — | — | — | — | — | CN21PUE | CN20PUE | CN19PUE | CN18PUE | CN17PUE | CN16PUE | 0000 |

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

**TABLE 4-4: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXGPX06A DEVICES**

| SFR Name | SFR 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CNEN1 | 0060 | CN15IE | CN14IE | CN13IE | CN12IE | CN11IE | CN10IE | CN9IE | CN8IE | CN7IE | CN6IE | CN5IE | CN4IE | CN3IE | CN2IE | CN1IE | CN0IE | 0000 |
| CNEN2 | 0062 | — | — | — | — | — | — | — | — | — | — | CN21IE | CN20IE | — | CN18IE | CN17IE | CN16IE | 0000 |
| CNPU1 | 0068 | CN15PUE | CN14PUE | CN13PUE | CN12PUE | CN11PUE | CN10PUE | CN9PUE | CN8PUE | CN7PUE | CN6PUE | CN5PUE | CN4PUE | CN3PUE | CN2PUE | CN1PUE | CN0PUE | 0000 |
| CNPU2 | 006A | — | — | — | — | — | — | — | — | — | — | CN21PUE | CN20PUE | — | CN18PUE | CN17PUE | CN16PUE | 0000 |

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

**TABLE 4-18: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 0 OR 1 FOR dsPIC33FJXXXGP506A/51A0/706A/708A/710A DEVICES ONLY**

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| C1CTRL1 | 0400 | — | — | CSIDL | ABAT | — | REQOP<2:0> | | | OPMODE<2:0> | | | — | CANCAP | — | — | WIN | 0480 |
| C1CTRL2 | 0402 | — | — | — | — | — | — | — | — | — | — | — | DNCNT<4:0> | | | | | 0000 |
| C1VEC | 0404 | — | — | — | FILHIT<4:0> | | | | | — | ICODE<6:0> | | | | | | | 0000 |
| C1FCTRL | 0406 | DMABS<2:0> | | | — | — | — | — | — | — | — | — | FSA<4:0> | | | | | 0000 |
| C1FIFO | 0408 | — | — | FBP<5:0> | | | | | | — | — | FNRB<5:0> | | | | | | 0000 |
| C1INTF | 040A | — | — | TXBO | TXBP | RXBP | TXWAR | RXWAR | EWARN | IVRIF | WAKIF | ERRIF | — | FIFOIF | RBOVIF | RBIF | TBIF | 0000 |
| C1INTE | 040C | — | — | — | — | — | — | — | — | IVRIE | WAKIE | ERRIE | — | FIFOIE | RBOVIE | RBIE | TBIE | 0000 |
| C1EC | 040E | TERRCNT<7:0> | | | | | | | | RERRCNT<7:0> | | | | | | | | 0000 |
| C1CFG1 | 0410 | — | — | — | — | — | — | — | — | SJW<1:0> | | BRP<5:0> | | | | | | 0000 |
| C1CFG2 | 0412 | — | WAKFIL | — | — | — | SEG2PH<2:0> | | | SEG2PHTS | SAM | SEG1PH<2:0> | | | PRSEG<2:0> | | | 0000 |
| C1FEN1 | 0414 | FLTEN15 | FLTEN14 | FLTEN13 | FLTEN12 | FLTEN11 | FLTEN10 | FLTEN9 | FLTEN8 | FLTEN7 | FLTEN6 | FLTEN5 | FLTEN4 | FLTEN3 | FLTEN2 | FLTEN1 | FLTEN0 | FFFF |
| C1FMSKSEL1 | 0418 | F7MSK<1:0> | | F6MSK<1:0> | | F5MSK<1:0> | | F4MSK<1:0> | | F3MSK<1:0> | | F2MSK<1:0> | | F1MSK<1:0> | | F0MSK<1:0> | | 0000 |
| C1FMSKSEL2 | 041A | F15MSK<1:0> | | F14MSK<1:0> | | F13MSK<1:0> | | F12MSK<1:0> | | F11MSK<1:0> | | F10MSK<1:0> | | F9MSK<1:0> | | F8MSK<1:0> | | 0000 |

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

**TABLE 4-19: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 0 FOR dsPIC33FJXXXGP506A/510A/706A/708A/710A DEVICES ONLY**

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0400-041E | **See definition when WIN = x** | | | | | | | | | | | | | | | | |
| C1RXFUL1 | 0420 | RXFUL15 | RXFUL14 | RXFUL13 | RXFUL12 | RXFUL11 | RXFUL10 | RXFUL9 | RXFUL8 | RXFUL7 | RXFUL6 | RXFUL5 | RXFUL4 | RXFUL3 | RXFUL2 | RXFUL1 | RXFUL0 | 0000 |
| C1RXFUL2 | 0422 | RXFUL31 | RXFUL30 | RXFUL29 | RXFUL28 | RXFUL27 | RXFUL26 | RXFUL25 | RXFUL24 | RXFUL23 | RXFUL22 | RXFUL21 | RXFUL20 | RXFUL19 | RXFUL18 | RXFUL17 | RXFUL16 | 0000 |
| C1RXOVF1 | 0428 | RXOVF15 | RXOVF14 | RXOVF13 | RXOVF12 | RXOVF11 | RXOVF10 | RXOVF9 | RXOVF8 | RXOVF7 | RXOVF6 | RXOVF5 | RXOVF4 | RXOVF3 | RXOVF2 | RXOVF1 | RXOVF0 | 0000 |
| C1RXOVF2 | 042A | RXOVF31 | RXOVF30 | RXOVF29 | RXOVF28 | RXOVF27 | RXOVF26 | RXOVF25 | RXOVF24 | RXOVF23 | RXOVF22 | RXOVF21 | RXOVF20 | RXOVF19 | RXOVF18 | RXOVF17 | RXOVF16 | 0000 |
| C1TR01CON | 0430 | TXEN1 | TXABT1 | TXLARB1 | TXERR1 | TXREQ1 | RTREN1 | TX1PRI<1:0> | | TXEN0 | TXABAT0 | TXLARB0 | TXERR0 | TXREQ0 | RTREN0 | TX0PRI<1:0> | | 0000 |
| C1TR23CON | 0432 | TXEN3 | TXABT3 | TXLARB3 | TXERR3 | TXREQ3 | RTREN3 | TX3PRI<1:0> | | TXEN2 | TXABAT2 | TXLARB2 | TXERR2 | TXREQ2 | RTREN2 | TX2PRI<1:0> | | 0000 |
| C1TR45CON | 0434 | TXEN5 | TXABT5 | TXLARB5 | TXERR5 | TXREQ5 | RTREN5 | TX5PRI<1:0> | | TXEN4 | TXABAT4 | TXLARB4 | TXERR4 | TXREQ4 | RTREN4 | TX4PRI<1:0> | | 0000 |
| C1TR67CON | 0436 | TXEN7 | TXABT7 | TXLARB7 | TXERR7 | TXREQ7 | RTREN7 | TX7PRI<1:0> | | TXEN6 | TXABAT6 | TXLARB6 | TXERR6 | TXREQ6 | RTREN6 | TX6PRI<1:0> | | xxxx |
| C1RXD | 0440 | Received Data Word | | | | | | | | | | | | | | | | xxxx |
| C1TXD | 0442 | Transmit Data Word | | | | | | | | | | | | | | | | xxxx |

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

The length of a circular buffer is not directly specified. It is determined by the difference between the corresponding start and end addresses. The maximum possible length of the circular buffer is 32K words (64 Kbytes).

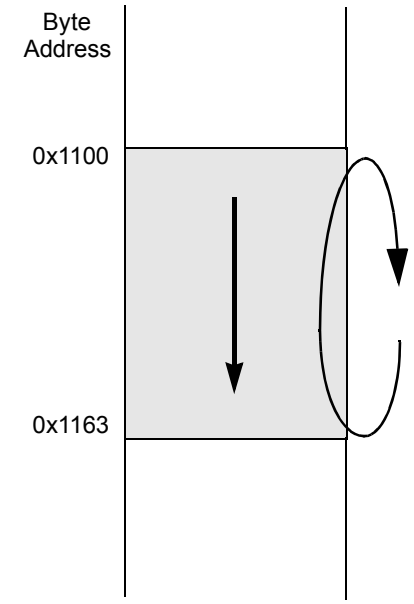
### 4.4.2 W ADDRESS REGISTER SELECTION

The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags as well as a W register field to specify the W Address registers. The XWM and YWM fields select which registers will operate with Modulo Addressing. If XWM = 15, X RAGU and X WAGU Modulo Addressing is disabled. Similarly, if YWM = 15, Y AGU Modulo Addressing is disabled.

The X Address Space Pointer W register (XWM), to which Modulo Addressing is to be applied, is stored in MODCON<3:0> (see Table 4-1). Modulo Addressing is enabled for X data space when XWM is set to any value other than '15' and the XMODEN bit is set at MODCON<15>.

The Y Address Space Pointer W register (YWM) to which Modulo Addressing is to be applied is stored in MODCON<7:4>. Modulo Addressing is enabled for Y data space when YWM is set to any value other than '15' and the YMODEN bit is set at MODCON<14>.

**FIGURE 4-7: MODULO ADDRESSING OPERATION EXAMPLE**




```
MOV     #0x1100, W0
MOV     W0, XMODSRT     ;set modulo start address
MOV     #0x1163, W0
MOV     W0, MODEND      ;set modulo end address
MOV     #0x8001, W0
MOV     W0, MODCON      ;enable W1, X AGU for modulo

MOV     #0x0000, W0     ;W0 holds buffer fill value

MOV     #0x1110, W1     ;point W1 to buffer

DO      AGAIN, #0x31    ;fill the 50 buffer locations
MOV     W0, [W1++]      ;fill the next location
AGAIN:  INC  W0, W0     ;increment the fill value
```

## 7.0 INTERRUPT CONTROLLER

**Note 1:** This data sheet summarizes the features of the dsPIC33FJXXXGPX06A/X08A/X10A 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 6. "Interrupts"** (DS70184) in the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

**2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXGPX06A/X08A/X10A interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33FJXXXGPX06A/X08A/X10A CPU. It has the following features:

- Up to eight processor exceptions and software traps
- Seven user-selectable priority levels
- Interrupt Vector Table (IVT) with up to 118 vectors
- A unique vector for each interrupt or exception source
- Fixed priority within a specified user priority level
- Alternate Interrupt Vector Table (AIVT) for debug support
- Fixed interrupt entry and return latencies

### 7.1 Interrupt Vector Table

The Interrupt Vector Table is shown in Figure 7-1. The IVT resides in program memory, starting at location 000004h. The IVT contains 126 vectors consisting of eight non-maskable trap vectors plus up to 118 sources of interrupt. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority; this priority is linked to their position in the vector table. All other things being equal, lower addresses have a higher natural priority. For example, the interrupt associated with vector 0 will take priority over interrupts at any other vector address.

dsPIC33FJXXXGPX06A/X08A/X10A devices implement up to 67 unique interrupts and five non-maskable traps. These are summarized in Table 7-1 and Table 7-2.

#### 7.1.1 ALTERNATE VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT) is located after the IVT, as shown in Figure 7-1. Access to the AIVT is provided by the ALTIVT control bit (INTCON2<15>). If the ALTIVT bit is set, all interrupt and exception processes use the alternate vectors instead of the default vectors. The alternate vectors are organized in the same manner as the default vectors.

The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time. If the AIVT is not needed, the AIVT should be programmed with the same addresses used in the IVT.

### 7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33FJXXXGPX06A/X08A/X10A device clears its registers in response to a Reset, which forces the PC to zero. The digital signal controller then begins program execution at location 0x000000. The user programs a `GOTO` instruction at the Reset address which redirects program execution to the appropriate start-up routine.

**Note:** Any unimplemented or unused vector locations in the IVT and AIVT should be programmed with the address of a default interrupt handler routine that contains a `RESET` instruction.

**REGISTER 7-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| U2TXIF | U2RXIF | INT2IF | T5IF | T4IF | OC4IF | OC3IF | DMA21IF |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| IC8IF | IC7IF | AD2IF | INT1IF | CNIF | — | MI2C1IF | SI2C1IF |
| 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 **U2TXIF:** UART2 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 14 **U2RXIF:** UART2 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **INT2IF:** External Interrupt 2 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12 **T5IF:** Timer5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 11 **T4IF:** Timer4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 10 **OC4IF:** Output Compare Channel 4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **OC3IF:** Output Compare Channel 3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **DMA21IF:** DMA Channel 2 Data Transfer Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **IC8IF:** Input Capture Channel 8 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **IC7IF:** Input Capture Channel 7 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **AD2IF:** ADC2 Conversion Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **INT1IF:** External Interrupt 1 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

**REGISTER 7-9: IFS4: INTERRUPT FLAG STATUS REGISTER 4**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | U-0 |
|---|---|---|---|---|---|---|---|
| C2TXIF | C1TXIF | DMA7IF | DMA6IF | — | U2EIF | U1EIF | — |
| 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 15-8 **Unimplemented:** Read as '0'

bit 7 **C2TXIF:** ECAN2 Transmit Data Request Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **C1TXIF:** ECAN1 Transmit Data Request Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **DMA7IF:** DMA Channel 7 Data Transfer Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **DMA6IF:** DMA Channel 6 Data Transfer Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

bit 2 **U2EIF:** UART2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **U1EIF:** UART1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

**REGISTER 7-29: IPC14: INTERRUPT PRIORITY CONTROL REGISTER 14**

| U-0 | R/W-1 | R/W-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | DCIEIP<2:0> | | | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-1 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | C2IP<2:0> | | |
| 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 14-12 **DCIEIP<2:0>:** DCI Error Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

bit 11-3 **Unimplemented:** Read as '0'

bit 2-0 **C2IP<2:0>:** ECAN2 Event Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

## 9.1 CPU Clocking System

There are seven system clock options provided by the dsPIC33FJXXXGPX06A/X08A/X10A:

- FRC Oscillator
- FRC Oscillator with PLL
- Primary (XT, HS or EC) Oscillator
- Primary Oscillator with PLL
- Secondary (LP) Oscillator
- LPRC Oscillator
- FRC Oscillator with postscaler

### 9.1.1 SYSTEM CLOCK SOURCES

The FRC (Fast RC) internal oscillator runs at a nominal frequency of 7.37 MHz. The user software can tune the FRC frequency. User software can optionally specify a factor (ranging from 1:2 to 1:256) by which the FRC clock frequency is divided. This factor is selected using the FRCDIV<2:0> (CLKDIV<10:8>) bits.

The primary oscillator can use one of the following as its clock source:

- XT (Crystal): Crystals and ceramic resonators in the range of 3 MHz to 10 MHz. The crystal is connected to the OSC1 and OSC2 pins.
- HS (High-Speed Crystal): Crystals in the range of 10 MHz to 40 MHz. The crystal is connected to the OSC1 and OSC2 pins.
- EC (External Clock): External clock signal is directly applied to the OSC1 pin.

The secondary (LP) oscillator is designed for low power and uses a 32.768 kHz crystal or ceramic resonator. The LP oscillator uses the SOSCI and SOSCO pins.

The LPRC (Low-Power RC) internal oscillator runs at a nominal frequency of 32.768 kHz. It is also used as a reference clock by the Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

The clock signals generated by the FRC and primary oscillators can be optionally applied to an on-chip Phase-Locked Loop (PLL) to provide a wide range of output frequencies for device operation. PLL configuration is described in **Section 9.1.3 "PLL Configuration"**.

The FRC frequency depends on the FRC accuracy (see Table 25-19) and the value of the FRC Oscillator Tuning register (see Register 9-4).

### 9.1.2 SYSTEM CLOCK SELECTION

The oscillator source that is used at a device Power-on Reset event is selected using Configuration bit settings. The oscillator Configuration bit settings are located in the Configuration registers in the program memory. (Refer to **Section 22.1 "Configuration Bits"** for further details.) The Initial Oscillator Selection Configuration bits, FNOSC<2:0> (FOSCSEL<2:0>), and the Primary Oscillator Mode Select Configuration bits, POSCMD<1:0> (FOSC<1:0>), select the oscillator source that is used at a Power-on Reset. The FRC primary oscillator is the default (unprogrammed) selection.

The Configuration bits allow users to choose between twelve different clock modes, shown in Table 9-1.

The output of the oscillator (or the output of the PLL if a PLL mode has been selected) FOSC is divided by 2 to generate the device instruction clock (FCY) and the peripheral clock time base (FP). FCY defines the operating speed of the device, and speeds up to 40 MHz are supported by the dsPIC33FJXXXGPX06A/X08A/X10A architecture.

Instruction execution speed or device operating frequency, FCY, is given by:

**EQUATION 9-1: DEVICE OPERATING FREQUENCY**

$$FCY = \frac{FOSC}{2}$$

### 9.1.3 PLL CONFIGURATION

The primary oscillator and internal FRC oscillator can optionally use an on-chip PLL to obtain higher speeds of operation. The PLL provides a significant amount of flexibility in selecting the device operating speed. A block diagram of the PLL is shown in Figure 9-2.

The output of the primary oscillator or FRC, denoted as 'FIN', is divided down by a prescale factor (N1) of 2, 3, ... or 33 before being provided to the PLL's Voltage Controlled Oscillator (VCO). The input to the VCO must be selected to be in the range of 0.8 MHz to 8 MHz. Since the minimum prescale factor is 2, this implies that FIN must be chosen to be in the range of 1.6 MHz to 16 MHz. The prescale factor 'N1' is selected using the PLLPRE<4:0> bits (CLKDIV<4:0>).

The PLL Feedback Divisor, selected using the PLLDIV<8:0> bits (PLLFBD<8:0>), provides a factor 'M', by which the input to the VCO is multiplied. This factor must be selected such that the resulting VCO output frequency is in the range of 100 MHz to 200 MHz.

The VCO output is further divided by a postscale factor 'N2'. This factor is selected using the PLLPOST<1:0> bits (CLKDIV<7:6>). 'N2' can be either 2, 4 or 8, and must be selected such that the PLL output frequency (FOSC) is in the range of 12.5 MHz to 80 MHz, which generates device operating speeds of 6.25-40 MIPS.

For a primary oscillator or FRC oscillator, output 'FIN', the PLL output 'FOSC' is given by:

**EQUATION 9-2: FOSC CALCULATION**

$$FOSC = FIN \cdot \left(\frac{M}{N1 \cdot N2}\right)$$

**REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER(1,3)**

| 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>(2) | | |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | U-0 | R-0 | U-0 | R/C-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| CLKLOCK | — | LOCK | — | CF | — | LPOSCEN | OSWEN |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | y = Value set from Configuration bits on POR | | C = Clear only 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 **Unimplemented:** Read as '0'

bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits (read-only)
111 = Fast RC oscillator (FRC) with Divide-by-N
110 = Fast RC oscillator (FRC) with Divide-by-16
101 = Low-Power RC oscillator (LPRC)
100 = Secondary oscillator (Sosc)
011 = Primary oscillator (XT, HS, EC) with PLL
010 = Primary oscillator (XT, HS, EC)
001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCDIVN + PLL)
000 = Fast RC oscillator (FRC)

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

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits(2)
111 = Fast RC oscillator (FRC) with Divide-by-N
110 = Fast RC oscillator (FRC) with Divide-by-16
101 = Low-Power RC oscillator (LPRC)
100 = Secondary oscillator (Sosc)
011 = Primary oscillator (XT, HS, EC) with PLL
010 = Primary oscillator (XT, HS, EC)
001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCDIVN + PLL)
000 = Fast RC oscillator (FRC)

bit 7 **CLKLOCK:** Clock Lock Enable bit
1 = If (FCKSM0 = 1), then clock and PLL configurations are locked
If (FCKSM0 = 0), then clock and PLL configurations may be modified
0 = Clock and PLL selections are not locked, configurations may be modified

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

bit 5 **LOCK:** PLL Lock Status bit (read-only)
1 = Indicates that PLL is in lock, or PLL start-up timer is satisfied
0 = Indicates that PLL is out of lock, start-up timer is in progress or PLL is disabled

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

bit 3 **CF:** Clock Fail Detect bit (read/clear by application)
1 = FSCM has detected clock failure
0 = FSCM has not detected clock failure

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

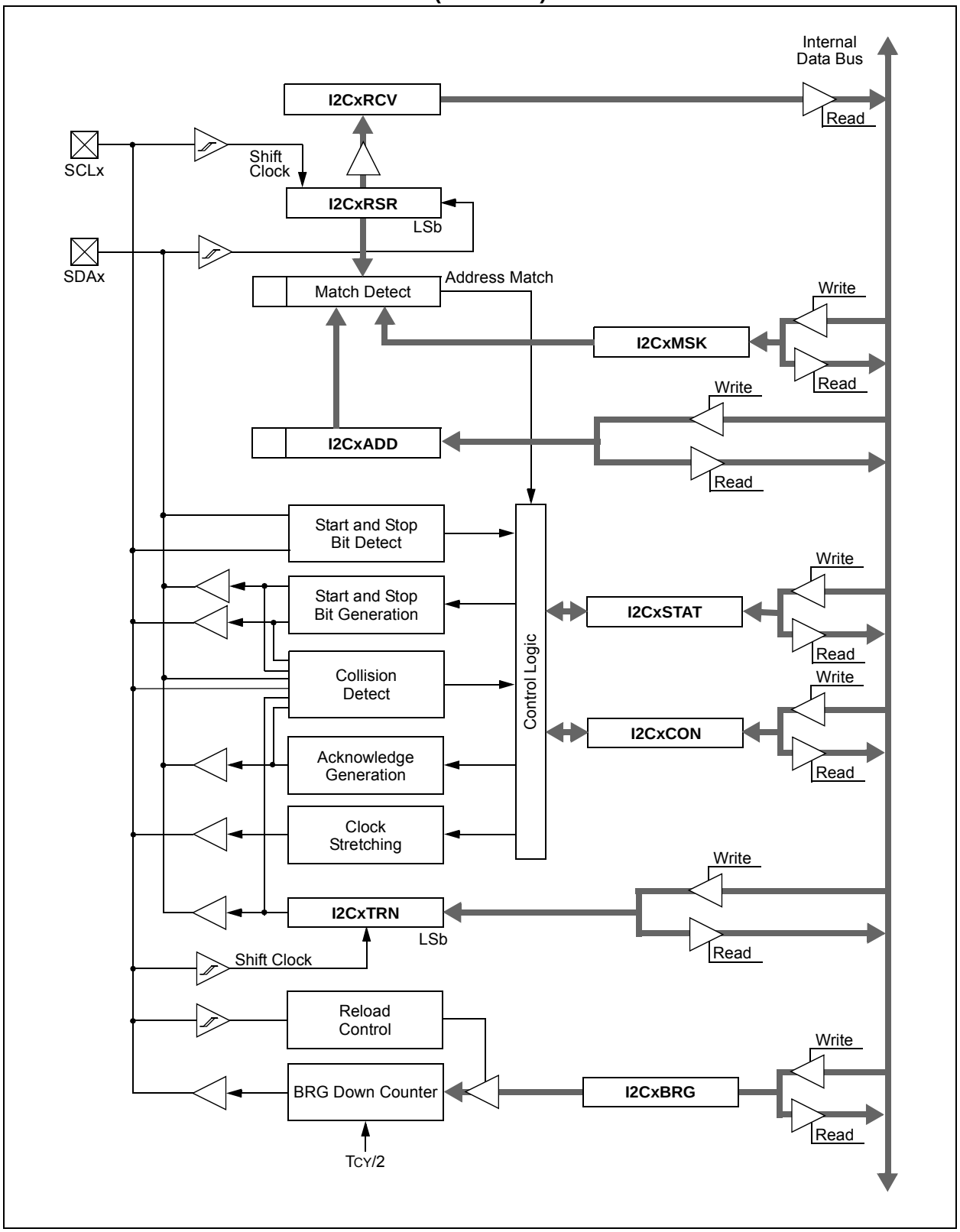
**Note 1:** Writes to this register require an unlock sequence. Refer to **Section 7. "Oscillator"** (DS70186) in the *"dsPIC33F/PIC24H Family Reference Manual"* for details.

**2:** Direct clock switches between any primary oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transition clock source between the two PLL modes.

**3:** This is register is reset only on a Power-on Reset (POR).

**NOTES:**

**FIGURE 17-1: I²C™ BLOCK DIAGRAM (X = 1 OR 2)**



## 19.3 Modes of Operation

The CAN module can operate in one of several operation modes selected by the user. These modes include:

- Initialization Mode
- Disable Mode
- Normal Operation Mode
- Listen Only Mode
- Listen All Messages Mode
- Loopback Mode

Modes are requested by setting the REQOP<2:0> bits (CiCTRL1<10:8>). Entry into a mode is Acknowledged by monitoring the OPMODE<2:0> bits (CiCTRL1<7:5>). The module will not change the mode and the OPMODE bits until a change in mode is acceptable, generally during bus Idle time, which is defined as at least 11 consecutive recessive bits.

### 19.3.1 INITIALIZATION MODE

In the Initialization mode, the module will not transmit or receive. The error counters are cleared and the interrupt flags remain unchanged. The programmer will have access to Configuration registers that are access restricted in other modes. The module will protect the user from accidentally violating the CAN protocol through programming errors. All registers which control the configuration of the module cannot be modified while the module is on-line. The CAN module will not be allowed to enter the Configuration mode while a transmission is taking place. The Configuration mode serves as a lock to protect the following registers:

- All Module Control Registers
- Baud Rate and Interrupt Configuration Registers
- Bus Timing Registers
- Identifier Acceptance Filter Registers
- Identifier Acceptance Mask Registers

### 19.3.2 DISABLE MODE

In Disable mode, the module will not transmit or receive. The module has the ability to set the WAKIF bit due to bus activity, however, any pending interrupts will remain and the error counters will retain their value.

If the REQOP<2:0> bits (CiCTRL1<10:8>) = 001, the module will enter the Module Disable mode. If the module is active, the module will wait for 11 recessive bits on the CAN bus, detect that condition as an Idle bus, then accept the module disable command. When the OPMODE<2:0> bits (CiCTRL1<7:5>) = 001, that indicates whether the module successfully went into Module Disable mode. The I/O pins will revert to normal I/O function when the module is in the Module Disable mode.

The module can be programmed to apply a low-pass filter function to the CiRX input line while the module or the CPU is in Sleep mode. The WAKFIL bit (CiCFG2<14>) enables or disables the filter.

| | |
|---|---|
| **Note:** | Typically, if the CAN module is allowed to transmit in a particular mode of operation and a transmission is requested immediately after the CAN module has been placed in that mode of operation, the module waits for 11 consecutive recessive bits on the bus before starting transmission. If the user switches to Disable mode within this 11-bit period, then this transmission is aborted and the corresponding TXABT bit is set and TXREQ bit is cleared. |

### 19.3.3 NORMAL OPERATION MODE

Normal Operation mode is selected when REQOP<2:0> = 000. In this mode, the module is activated and the I/O pins will assume the CAN bus functions. The module will transmit and receive CAN bus messages via the CiTX and CiRX pins.

### 19.3.4 LISTEN ONLY MODE

If the Listen Only mode is activated, the module on the CAN bus is passive. The transmitter buffers revert to the port I/O function. The receive pins remain inputs. For the receiver, no error flags or Acknowledge signals are sent. The error counters are deactivated in this state. The Listen Only mode can be used for detecting the baud rate on the CAN bus. To use this, it is necessary that there are at least two further nodes that communicate with each other.

### 19.3.5 LISTEN ALL MESSAGES MODE

The module can be set to ignore all errors and receive any message. The Listen All Messages mode is activated by setting REQOP<2:0> = '111'. In this mode, the data which is in the message assembly buffer, until the time an error occurred, is copied in the receive buffer and can be read via the CPU interface.

### 19.3.6 LOOPBACK MODE

If the Loopback mode is activated, the module will connect the internal transmit signal to the internal receive signal at the module boundary. The transmit and receive pins revert to their port I/O function.

**REGISTER 19-16: CiRXFnSID: ECAN™ ACCEPTANCE FILTER n STANDARD IDENTIFIER (n = 0, 1, ..., 15)**

| R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| SID<10:3> | | | | | | | |
| bit 15 | | | | | | | bit 8 |

| R/W-x | R/W-x | R/W-x | U-0 | R/W-x | U-0 | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| SID<2:0> | | | — | EXIDE | — | EID<17:16> | |
| 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 15-5 **SID<10:0>:** Standard Identifier bits
1 = Message address bit SIDx must be '1' to match filter
0 = Message address bit SIDx must be '0' to match filter

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

bit 3 **EXIDE:** Extended Identifier Enable bit
If MIDE = 1 then:
1 = Match only messages with extended identifier addresses
0 = Match only messages with standard identifier addresses
If MIDE = 0 then:
Ignore EXIDE bit.

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

bit 1-0 **EID<17:16>:** Extended Identifier bits
1 = Message address bit EIDx must be '1' to match filter
0 = Message address bit EIDx must be '0' to match filter

**REGISTER 19-17: CiRXFnEID: ECAN™ ACCEPTANCE FILTER n EXTENDED IDENTIFIER (n = 0, 1, ..., 15)**

| R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| EID<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 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 |
|---|---|---|---|---|---|---|---|
| EID<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 **EID<15:0>:** Extended Identifier bits
1 = Message address bit EIDx must be '1' to match filter
0 = Message address bit EIDx must be '0' to match filter

**TABLE 22-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)**

| Bit Field | Register | RTSP Effect | Description |
|---|---|---|---|
| SSS<2:0> | FSS | Immediate | Secure Segment Program Flash Code Protection Size<br><br>**(FOR 128K and 256K DEVICES)**<br>X11 = No Secure program Flash segment<br><br>Secure space is 8K IW less BS<br>110 = Standard security; secure program Flash segment starts at End of BS, ends at 0x003FFE<br>010 = High security; secure program Flash segment starts at End of BS, ends at 0x003FFE<br><br>Secure space is 16K IW less BS<br>101 = Standard security; secure program Flash segment starts at End of BS, ends at 0x007FFE<br>001 = High security; secure program Flash segment starts at End of BS, ends at 0x007FFE<br><br>Secure space is 32K IW less BS<br>100 = Standard security; secure program Flash segment starts at End of BS, ends at 0x00FFFE<br>000 = High security; secure program Flash segment starts at End of BS, ends at 0x00FFFE<br><br>**(FOR 64K DEVICES)**<br>X11 = No Secure program Flash segment<br><br>Secure space is 4K IW less BS<br>110 = Standard security; secure program Flash segment starts at End of BS, ends at 0x001FFE<br>010 = High security; secure program Flash segment starts at End of BS, ends at 0x001FFE<br><br>Secure space is 8K IW less BS<br>101 = Standard security; secure program Flash segment starts at End of BS, ends at 0x003FFE<br>001 = High security; secure program Flash segment starts at End of BS, ends at 0x003FFE<br><br>Secure space is 16K IW less BS<br>100 = Standard security; secure program Flash segment starts at End of BS, ends at 007FFEh<br>000 = High security; secure program Flash segment starts at End of BS, ends at 0x007FFE |
| RSS<1:0> | FSS | Immediate | Secure Segment RAM Code Protection<br>11 = No Secure RAM defined<br>10 = Secure RAM is 256 Bytes less BS RAM<br>01 = Secure RAM is 2048 Bytes less BS RAM<br>00 = Secure RAM is 4096 Bytes less BS RAM |
| GSS<1:0> | FGS | Immediate | General Segment Code-Protect bit<br>11 = User program memory is not code-protected<br>10 = Standard security; general program Flash segment starts at End of SS, ends at EOM<br>0x = High security; general program Flash segment starts at End of SS, ends at EOM |

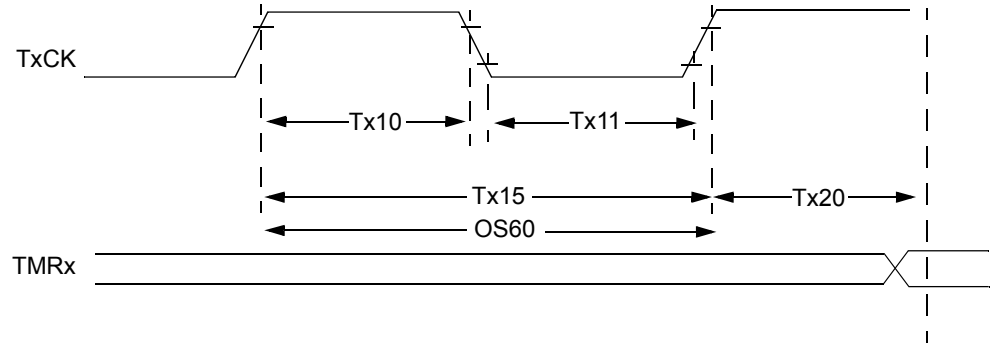
**TABLE 25-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS**

| DC CHARACTERISTICS | | | Standard Operating Conditions: 3.0V 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** | **Characteristic** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| **Operating Voltage** | | | | | | | |
| DC10 | **Supply Voltage** | | | | | | |
| | VDD | — | 3.0 | — | 3.6 | V | — |
| DC12 | VDR | **RAM Data Retention Voltage(2)** | 1.8 | — | — | V | — |
| DC16 | VPOR | **VDD Start Voltage** to ensure internal Power-on Reset signal | — | — | VSS | V | — |
| DC17 | SVDD | **VDD Rise Rate** to ensure internal Power-on Reset signal | 0.03 | — | — | V/ms | 0-3.0V in 0.1s |

**Note 1:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

**2:** This is the limit to which VDD can be lowered without losing RAM data.

**FIGURE 25-5: TIMER1, 2, 3, 4, 5, 6, 7, 8 AND 9 EXTERNAL CLOCK TIMING CHARACTERISTICS**



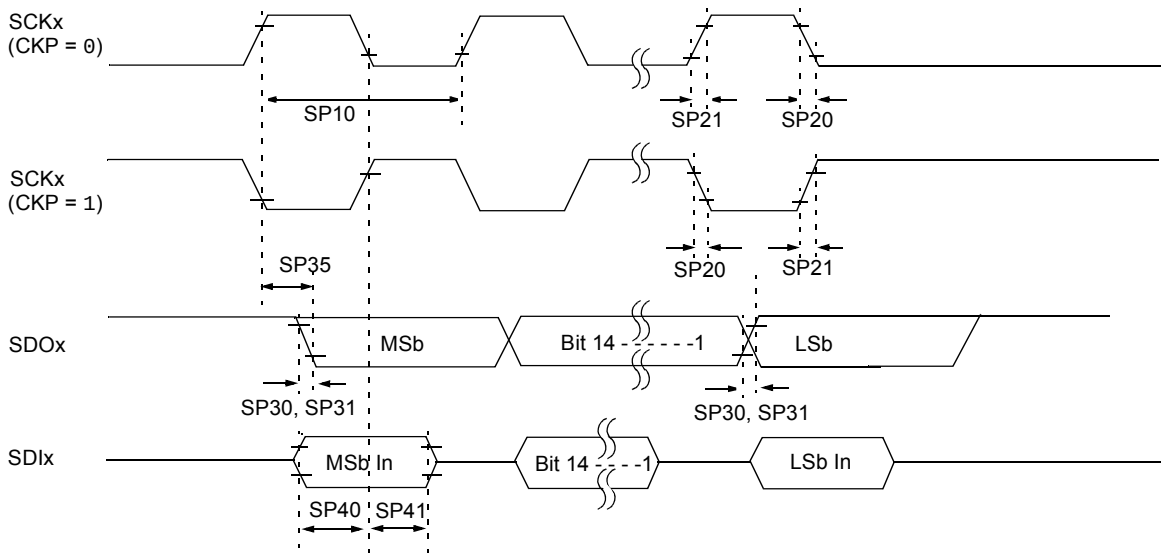

**Note:** Refer to Figure 25-1 for load conditions.

**TABLE 25-22: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS(1)**

| AC CHARACTERISTICS | | | | Standard Operating Conditions: 3.0V 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** | **Characteristic** | | **Min** | **Typ** | **Max** | **Units** | **Conditions** |
| TA10 | TTXH | TxCK High Time | Synchronous, no prescaler | TCY + 20 | — | — | ns | Must also meet parameter TA15 |
| | | | Synchronous, with prescaler | (TCY + 20)/N | — | — | ns | |
| | | | Asynchronous | 20 | — | — | ns | |
| TA11 | TTXL | TxCK Low Time | Synchronous, no prescaler | (TCY + 20)/N | — | — | ns | Must also meet parameter TA15 |
| | | | Synchronous, with prescaler | 20 | — | — | ns | N = prescale value (1,8,64,256) |
| | | | Asynchronous | 20 | — | — | ns | |
| TA15 | TTXP | TxCK Input Period | Synchronous, no prescaler | 2TCY + 40 | — | — | ns | — |
| | | | Synchronous, with prescaler | Greater of 40 ns or (2TCY + 40)/N | — | — | — | N = prescale value (1, 8, 64, 256) |
| | | | Asynchronous | 40 | — | — | ns | — |
| OS60 | Ft1 | SOSC1/T1CK Oscillator Input frequency Range (oscillator enabled by setting TCS bit (T1CON<1>)) | | DC | — | 50 | kHz | — |
| TA20 | TCKEXTMRL | Delay from External TxCK Clock Edge to Timer Increment | | 0.75TCY+40 | — | 1.75TCY+40 | ns | — |

**Note 1:** Timer1 is a Type A.

**FIGURE 25-12: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = X, SMP = 1) TIMING CHARACTERISTICS**




**Note:** Refer to Figure 25-1 for load conditions.

**TABLE 25-31: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS**

| AC CHARACTERISTICS | | | Standard Operating Conditions: 2.4V 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** | **Characteristic(1)** | **Min** | **Typ(2)** | **Max** | **Units** | **Conditions** |
| SP10 | TscP | Maximum SCK Frequency | — | — | 10 | MHz | -40ºC to +125ºC and see **Note 3** |
| SP20 | TscF | SCKx Output Fall Time | — | — | — | ns | See parameter DO32 and **Note 4** |
| SP21 | TscR | SCKx Output Rise Time | — | — | — | ns | See parameter DO31 and **Note 4** |
| SP30 | TdoF | SDOx Data Output Fall Time | — | — | — | ns | See parameter DO32 and **Note 4** |
| SP31 | TdoR | SDOx Data Output Rise Time | — | — | — | ns | See parameter DO31 and **Note 4** |
| SP35 | TscH2doV, TscL2doV | SDOx Data Output Valid after SCKx Edge | — | 6 | 20 | ns | — |
| SP36 | TdoV2scH, TdoV2scL | SDOx Data Output Setup to First SCKx Edge | 30 | — | — | ns | — |
| SP40 | TdiV2scH, TdiV2scL | Setup Time of SDIx Data Input to SCKx Edge | 30 | — | — | ns | — |
| SP41 | TscH2diL, TscL2diL | Hold Time of SDIx Data Input to SCKx Edge | 30 | — | — | ns | — |

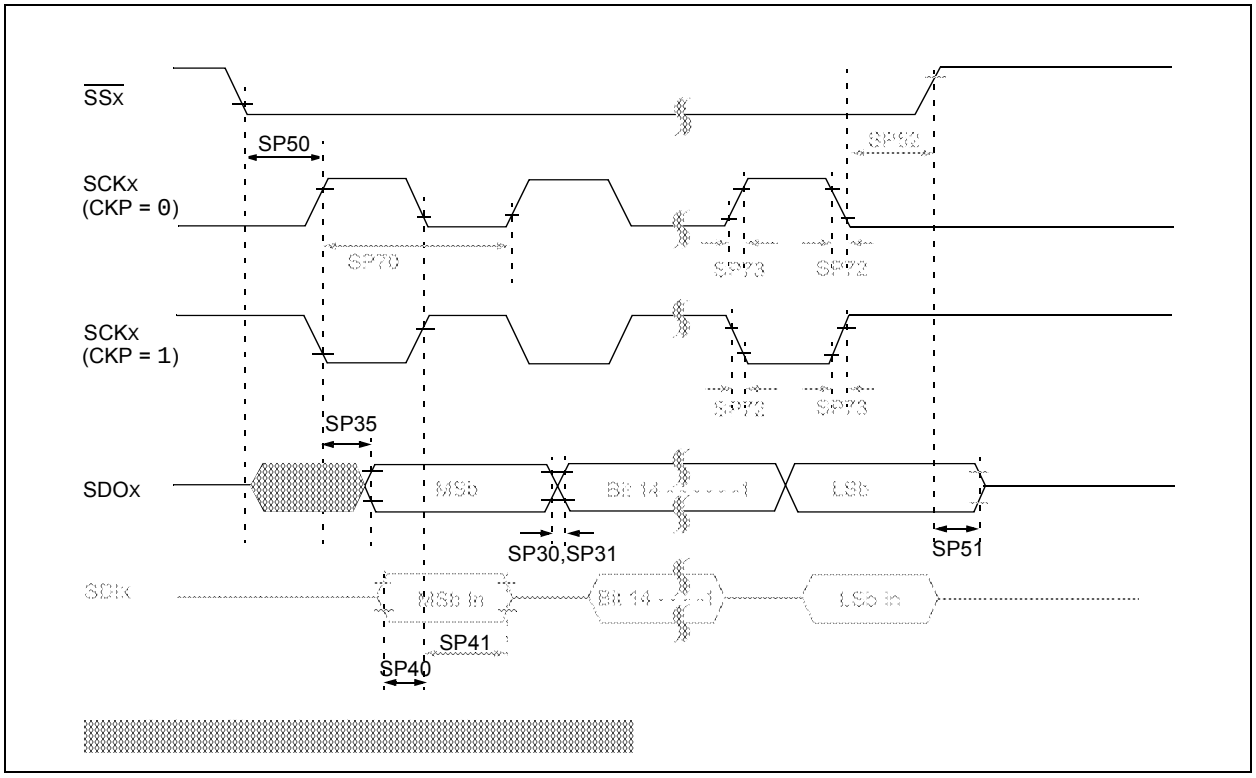
**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**2:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

**3:** The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.

**4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 25-16: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**



## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at 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 (FAQs), 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. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

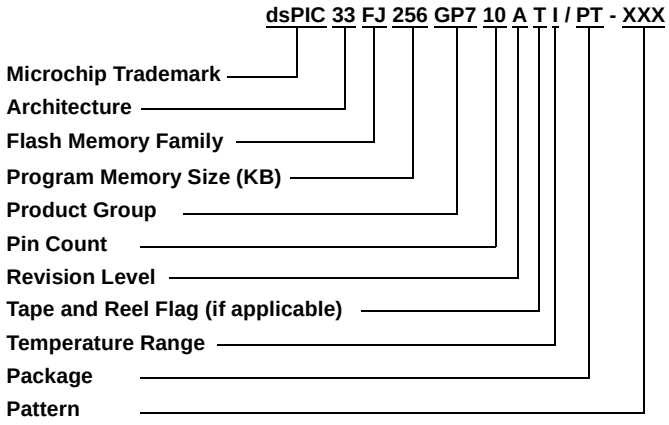
- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

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**

## PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.




| | | | |
|---|---|---|---|
| Architecture: | 33 | = | 16-bit Digital Signal Controller |
| Flash Memory Family: | FJ | = | Flash program memory, 3.3V |
| Product Group: | GP2 | = | General purpose family |
| | GP3 | = | General purpose family |
| | GP5 | = | General purpose family |
| | GP7 | = | General purpose family |
| Pin Count: | 06 | = | 64-pin |
| | 08 | = | 80-pin |
| | 10 | = | 100-pin |
| Temperature Range: | I | = | -40°C to+85°C(Industrial) |
| | E | = | -40°C to+125°C(Extended) |
| | H | = | -40°C to+150°C(High) |
| Package: | PT | = | 10x10 or 12x12 mm TQFP (Thin Quad Flatpack) |
| | PF | = | 14x14 mm TQFP (Thin Quad Flatpack) |
| | MR | = | 9x9mm QFN (Plastic Quad Flatpack) |
| Pattern | Three-digit QTP, SQTP, Code or Special Requirements (blank otherwise) | | |
| | ES | = | Engineering Sample |

**Examples:**

a) dsPIC33FJ256GP710AI/PT: General-purpose dsPIC33, 64 KB program memory, 100-pin, Industrial temp., TQFP package.