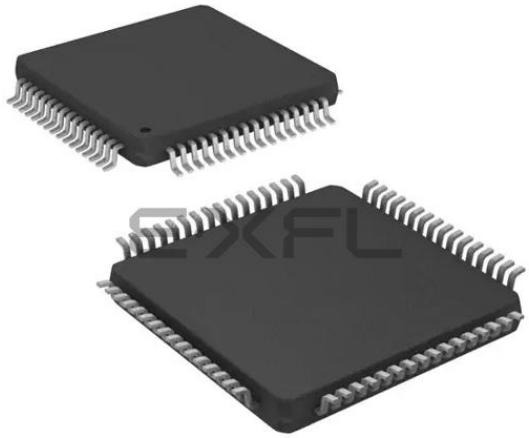
Microchip Technology - PIC24FJ64GA106T-I/PT Datasheet



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	Obsolete
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	53
Program Memory Size	64KB (22K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj64ga106t-i-pt

TABLE 4-4: ICN 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
CNPD1	0054	CN15PDE	CN14PDE	CN13PDE	CN12PDE	CN11PDE	CN10PDE	CN9PDE	CN8PDE	CN7PDE	CN6PDE	CN5PDE	CN4PDE	CN3PDE	CN2PDE	CN1PDE	CN0PDE	0000
CNPD2	0056	CN31PDE	CN30PDE	CN29PDE	CN28PDE	CN27PDE	CN26PDE	CN25PDE	CN24PDE	CN23PDE	CN22PDE	CN21PDE(1)	CN20PDE(1)	CN19PDE(1)	CN18PDE	CN17PDE	CN16PDE	0000
CNPD3	0058	CN47PDE(1)	CN46PDE(2)	CN45PDE(1)	CN44PDE(1)	CN43PDE(1)	CN42PDE(1)	CN41PDE(1)	CN40PDE(2)	CN39PDE(2)	CN38PDE(2)	CN37PDE(2)	CN36PDE(2)	CN35PDE(2)	CN34PDE(2)	CN33PDE(2)	CN32PDE	0000
CNPD4	005A	CN63PDE	CN62PDE	CN61PDE	CN60PDE	CN59PDE	CN58PDE	CN57PDE(1)	CN56PDE	CN55PDE	CN54PDE	CN53PDE	CN52PDE	CN51PDE	CN50PDE	CN49PDE	CN48PDE(2)	0000
CNPD5	005C	CN79PDE(2)	CN78PDE(1)	CN77PDE(1)	CN76PDE(2)	CN75PDE(2)	CN74PDE(1)	CN73PDE(1)	CN72PDE	CN71PDE	CN70PDE	CN69PDE	CN68PDE	CN67PDE(1)	CN66PDE(1)	CN65PDE	CN64PDE	0000
CNPD6	005E	—	—	—	—	—	—	—	—	—	—	—	CN84PDE	CN83PDE	CN82PDE(2)	CN81PDE(2)	CN80PDE(2)	0000
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	CN31IE	CN30IE	CN29IE	CN28IE	CN27IE	CN26IE	CN25IE	CN24IE	CN23IE	CN22IE	CN21IE(1)	CN20IE(1)	CN19IE(1)	CN18IE	CN17IE	CN16IE	0000
CNEN3	0064	CN47IE(1)	CN46IE(2)	CN45IE(1)	CN44IE(1)	CN43IE(1)	CN42IE(1)	CN41IE(1)	CN40IE(2)	CN39IE(2)	CN38IE(2)	CN37IE(2)	CN36IE(2)	CN35IE(2)	CN34IE(2)	CN33IE(2)	CN32IE	0000
CNEN4	0066	CN63IE	CN62IE	CN61IE	CN60IE	CN59IE	CN58IE	CN57IE(1)	CN56IE	CN55IE	CN54IE	CN53IE	CN52IE	CN51IE	CN50IE	CN49IE	CN48IE(2)	0000
CNEN5	0068	CN79IE(2)	CN78IE(1)	CN77IE(1)	CN76IE(2)	CN75IE(2)	CN74IE(1)	CN73IE(1)	CN72IE	CN71IE	CN70IE	CN69IE	CN68IE	CN67IE(1)	CN66IE(1)	CN65IE	CN64IE	0000
CNEN6	006A	—	—	—	—	—	—	—	—	—	—	—	CN84IE	CN83IE	CN82IE(2)	CN81IE(2)	CN80IE(2)	0000
CNPU1	006C	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006E	CN31PUE	CN30PUE	CN29PUE	CN28PUE	CN27PUE	CN26PUE	CN25PUE	CN24PUE	CN23PUE	CN22PUE	CN21PUE(1)	CN20PUE(1)	CN19PUE(1)	CN18PUE	CN17PUE	CN16PUE	0000
CNPU3	0070	CN47PUE(1)	CN46PUE(2)	CN45PUE(1)	CN44PUE(1)	CN43PUE(1)	CN42PUE(1)	CN41PUE(1)	CN40PUE(2)	CN39PUE(2)	CN38PUE(2)	CN37PUE(2)	CN36PUE(2)	CN35PUE(2)	CN34PUE(2)	CN33PUE(2)	CN32PUE	0000
CNPU4	0072	CN63PUE	CN62PUE	CN61PUE	CN60PUE	CN59PUE	CN58PUE	CN57PUE(1)	CN56PUE	CN55PUE	CN54PUE	CN53PUE	CN52PUE	CN51PUE	CN50PUE	CN49PUE	CN48PUE(2)	0000
CNPU5	0074	CN79PUE(2)	CN78PUE(1)	CN77PUE(1)	CN76PUE(2)	CN75PUE(2)	CN74PUE(1)	CN73PUE(1)	CN72PUE	CN71PUE	CN70PUE	CN69PUE	CN68PUE	CN67PUE(1)	CN66PUE(1)	CN65PUE	CN64PUE	0000
CNPU6	0076	—	—	—	—	—	—	—	—	—	—	—	CN84PUE	CN83PUE	CN82PUE(2)	CN81PUE(2)	CN80PUE(2)	0000

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

Note 1: Unimplemented in 64-pin devices; read as '0'.

2: Unimplemented in 64-pin and 80-pin devices; read as '0'.

5.6.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

The user can program one row of Flash program memory at a time. To do this, it is necessary to erase the 8-row erase block containing the desired row. The general process is as follows:

1. Read eight rows of program memory (512 instructions) and store in data RAM.
2. Update the program data in RAM with the desired new data.
3. Erase the block (see Example 5-1 for an implementation in assembler):
   a) Set the NVMOP bits (NVMCON<3:0>) to '0010' to configure for block erase. Set the ERASE (NVMCON<6>) and WREN (NVMCON<14>) bits.
   b) Write the starting address of the block to be erased into the TBLPAG and W registers.
   c) Write 55h to NVMKEY.
   d) Write AAh to NVMKEY.
   e) Set the WR bit (NVMCON<15>). The erase cycle begins and the CPU stalls for the duration of the erase cycle. When the erase is done, the WR bit is cleared automatically.
4. Write the first 64 instructions from data RAM into the program memory buffers (see Example 5-3 for the implementation in assembler).
5. Write the program block to Flash memory:
   a) Set the NVMOP bits to '0001' to configure for row programming. Clear the ERASE bit and set the WREN bit.
   b) Write 55h to NVMKEY.
   c) Write AAh to NVMKEY.
   d) Set the WR bit. The programming cycle begins and the CPU stalls for the duration of the write cycle. When the write to Flash memory is done, the WR bit is cleared automatically.
6. Repeat Steps 4 and 5, using the next available 64 instructions from the block in data RAM by incrementing the value in TBLPAG, until all 512 instructions are written back to Flash memory.

For protection against accidental operations, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be NOPs, as shown in Example 5-5.

Note:	The equivalent C code for these steps, prepared using Microchip's MPLAB C30 compiler and a specific library of built-in hardware functions, is shown in Examples 5-2, 5-4 and 5-6.

EXAMPLE 5-1: ERASING A PROGRAM MEMORY BLOCK (ASSEMBLY LANGUAGE CODE)

```
; Set up NVMCON for block erase operation
       MOV     #0x4042, W0                 ;
       MOV     W0, NVMCON                  ; Initialize NVMCON
; Init pointer to row to be ERASED
       MOV     #tblpage(PROG_ADDR), W0     ;
       MOV     W0, TBLPAG                  ; Initialize PM Page Boundary SFR
       MOV     #tbloffset(PROG_ADDR), W0   ; Initialize in-page EA[15:0] pointer
       TBLWTL  W0, [W0]                    ; Set base address of erase block
       DISI    #5                          ; Block all interrupts with priority <7
                                           ; for next 5 instructions
       MOV     #0x55, W0
       MOV     W0, NVMKEY                  ; Write the 55 key
       MOV     #0xAA, W1                   ;
       MOV     W1, NVMKEY                  ; Write the AA key
       BSET    NVMCON, #WR                 ; Start the erase sequence
       NOP                                 ; Insert two NOPs after the erase
       NOP                                 ; command is asserted
```

7.0 INTERRUPT CONTROLLER

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"PIC24F Family Reference Manual"*, **Section 8. "Interrupts"** (DS39707).

The PIC24F interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the PIC24F CPU. It has the following features:

- Up to 8 processor exceptions and software traps
- 7 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 (IVT) is shown in Figure 7-1. The IVT resides in program memory, starting at location 000004h. The IVT contains 126 vectors, consisting of 8 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 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.

PIC24FJ256GA110 family devices implement non-maskable traps and unique interrupts. These are summarized in Table 7-1 and Table 7-2.

7.1.1 ALTERNATE INTERRUPT 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 will 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 emulation and debugging efforts 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 PIC24F devices clear their registers in response to a Reset which forces the PC to zero. The microcontroller then begins program execution at location 000000h. 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 7-9: IFS4: INTERRUPT FLAG STATUS REGISTER 4

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
—	—	CTMUIF	—	—	—	—	LVDIF
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	—	CRCIF	U2ERIF	U1ERIF	—
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 15-14 **Unimplemented:** Read as '0'

bit 13 **CTMUIF:** CTMU Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12-9 **Unimplemented:** Read as '0'

bit 8 **LVDIF:** Low-Voltage Detect Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7-4 **Unimplemented:** Read as '0'

bit 3 **CRCIF:** CRC Generator Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **U2ERIF:** UART2 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **U1ERIF:** UART1 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

REGISTER 7-30: IPC13: INTERRUPT PRIORITY CONTROL REGISTER 13

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT4IP2	INT4IP1	INT4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	INT3IP2	INT3IP1	INT3IP0	—	—	—	—
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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **INT4IP<2:0>:** External Interrupt 4 Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

bit 6-4 **INT3IP<2:0>:** External Interrupt 3 Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

7.4 Interrupt Setup Procedures

7.4.1 INITIALIZATION

To configure an interrupt source:

1. Set the NSTDIS control bit (INTCON1<15>) if nested interrupts are not desired.
2. Select the user-assigned priority level for the interrupt source by writing the control bits in the appropriate IPCx register. The priority level will depend on the specific application and type of interrupt source. If multiple priority levels are not desired, the IPCx register control bits for all enabled interrupt sources may be programmed to the same non-zero value.

Note:	At a device Reset, the IPCx registers are initialized, such that all user interrupt sources are assigned to priority level 4.

3. Clear the interrupt status flag bit associated with the peripheral in the associated IFSx register.
4. Enable the interrupt source by setting the interrupt enable control bit associated with the source in the appropriate IECx register.

7.4.2 INTERRUPT SERVICE ROUTINE

The method that is used to declare an ISR and initialize the IVT with the correct vector address will depend on the programming language (i.e., 'C' or assembler) and the language development toolsuite that is used to develop the application. In general, the user must clear the interrupt flag in the appropriate IFSx register for the source of the interrupt that the ISR handles. Otherwise, the ISR will be re-entered immediately after exiting the routine. If the ISR is coded in assembly language, it must be terminated using a `RETFIE` instruction to unstack the saved PC value, SRL value and old CPU priority level.

7.4.3 TRAP SERVICE ROUTINE

A Trap Service Routine (TSR) is coded like an ISR, except that the appropriate trap status flag in the INTCON1 register must be cleared to avoid re-entry into the TSR.

7.4.4 INTERRUPT DISABLE

All user interrupts can be disabled using the following procedure:

1. Push the current SR value onto the software stack using the `PUSH` instruction.
2. Force the CPU to priority level 7 by inclusive ORing the value E0h with SRL.

To enable user interrupts, the `POP` instruction may be used to restore the previous SR value.

Note that only user interrupts with a priority level of 7 or less can be disabled. Trap sources (level 8-15) cannot be disabled.

The `DISI` instruction provides a convenient way to disable interrupts of priority levels 1-6 for a fixed period of time. Level 7 interrupt sources are not disabled by the `DISI` instruction.

REGISTER 8-3: OSCTUN: FRC OSCILLATOR TUNE REGISTER

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5(1)	TUN4(1)	TUN3(1)	TUN2(1)	TUN1(1)	TUN0(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 15-6 **Unimplemented:** Read as '0'

bit 5-0 **TUN<5:0>:** FRC Oscillator Tuning bits(1)

011111 = Maximum frequency deviation
011110 =
•
•
•
000001 =
000000 = Center frequency, oscillator is running at factory calibrated frequency
111111 =
•
•
•
100001 =
100000 = Minimum frequency deviation

Note 1: Increments or decrements of TUN<5:0> may not change the FRC frequency in equal steps over the FRC tuning range and may not be monotonic.

8.4 Clock Switching Operation

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC24F devices have a safeguard lock built into the switching process.

Note: The Primary Oscillator mode has three different submodes (XT, HS and EC) which are determined by the POSCMDx Configuration bits. While an application can switch to and from Primary Oscillator mode in software, it cannot switch between the different primary submodes without reprogramming the device.

8.4.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in CW 2 must be programmed to '0'. (Refer to **Section 25.1 "Configuration Bits"** for further details.) If the FCKSM1 Configuration bit is unprogrammed ('1'), the clock switching function and Fail-Safe Clock Monitor function are disabled; this is the default setting.

The NOSCx control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSCx bits (OSCCON<14:12>) will reflect the clock source selected by the FNOSCx Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled; it is held at '0' at all times.

REGISTER 10-21: RPINR29: PERIPHERAL PIN SELECT INPUT REGISTER 29

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

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	SS3R5	SS3R4	SS3R3	SS3R2	SS3R1	SS3R0
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 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SS3R<5:0>:** Assign SPI3 Slave Select Input (SS31IN) to Corresponding RPn or RPIn Pin bits

REGISTER 10-36: RPOR14: PERIPHERAL PIN SELECT OUTPUT REGISTER 14

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP29R5	RP29R4	RP29R3	RP29R2	RP29R1	RP29R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP28R5	RP28R4	RP28R3	RP28R2	RP28R1	RP28R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP29R<5:0>:** RP29 Output Pin Mapping bits
Peripheral output number n is assigned to pin, RP29 (see Table 10-3 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP28R<5:0>:** RP28 Output Pin Mapping bits
Peripheral output number n is assigned to pin, RP28 (see Table 10-3 for peripheral function numbers).

REGISTER 10-37: RPOR15: PERIPHERAL PIN SELECT OUTPUT REGISTER 15

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP31R5(1)	RP31R4(1)	RP31R3(1)	RP31R2(1)	RP31R1(1)	RP31R0(1)
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP30R5	RP30R4	RP30R3	RP30R2	RP30R1	RP30R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP31R<5:0>:** RP31 Output Pin Mapping bits(1)
Peripheral output number n is assigned to pin, RP31 (see Table 10-3 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP30R<5:0>:** RP30 Output Pin Mapping bits
Peripheral output number n is assigned to pin, RP30 (see Table 10-3 for peripheral function numbers).

Note 1: Unimplemented in 64-pin and 80-pin devices; read as '0'.

REGISTER 16-2: I2CxSTAT: I2Cx STATUS REGISTER

R-0, HSC	R-0, HSC	U-0	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0, HS	R/C-0, HS	R-0, HSC	R/C-0, HSC	R/C-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IWCOL	I2COV	D/$\overline{A}$	P	S	R/$\overline{W}$	RBF	TBF
bit 7							bit 0

Legend:	C = Clearable bit	HS = Hardware Settable bit	HSC = Hardware Settable/Clearable bit
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **ACKSTAT:** Acknowledge Status bit
1 = NACK was detected last
0 = ACK was detected last
Hardware set or clear at end of Acknowledge.

bit 14 **TRSTAT:** Transmit Status bit
(When operating as I^2C master. Applicable to master transmit operation.)
1 = Master transmit is in progress (8 bits + ACK)
0 = Master transmit is not in progress
Hardware set at beginning of master transmission. Hardware clear at end of slave Acknowledge.

bit 13-11 **Unimplemented:** Read as '0'

bit 10 **BCL:** Master Bus Collision Detect bit
1 = A bus collision has been detected during a master operation
0 = No collision
Hardware set at detection of bus collision.

bit 9 **GCSTAT:** General Call Status bit
1 = General call address was received
0 = General call address was not received
Hardware set when address matches general call address. Hardware clear at Stop detection.

bit 8 **ADD10:** 10-Bit Address Status bit
1 = 10-bit address was matched
0 = 10-bit address was not matched
Hardware set at match of 2nd byte of matched 10-bit address. Hardware clear at Stop detection.

bit 7 **IWCOL:** Write Collision Detect bit
1 = An attempt to write to the I2CxTRN register failed because the I^2C module is busy
0 = No collision
Hardware set at occurrence of write to I2CxTRN while busy (cleared by software).

bit 6 **I2COV:** Receive Overflow Flag bit
1 = A byte was received while the I2CxRCV register is still holding the previous byte
0 = No overflow
Hardware set at attempt to transfer I2CxRSR to I2CxRCV (cleared by software).

bit 5 **D/$\overline{A}$:** Data/$\overline{\text{Address}}$ bit (when operating as I^2C slave)
1 = Indicates that the last byte received was data
0 = Indicates that the last byte received was device address
Hardware clear at device address match. Hardware set after a transmission finishes or by reception of the slave byte.

REGISTER 18-6: PADCFG1: PAD CONFIGURATION CONTROL REGISTER

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	RTSECSEL(1)	PMPTTL
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 15-2 **Unimplemented:** Read as ‘0’

bit 1 **RTSECSEL:** RTCC Seconds Clock Output Select bit(1)
- 1 = RTCC seconds clock is selected for the RTCC pin
- 0 = RTCC alarm pulse is selected for the RTCC pin

bit 0 **PMPTTL:** PMP Module TTL Input Buffer Select bit
- 1 = PMP module inputs (PMDx, PMCS1) use TTL input buffers
- 0 = PMP module inputs use Schmitt Trigger input buffers

Note 1: To enable the actual RTCC output, the RTCOE (RCFGCAL<10>) bit must also be set.

FIGURE 18-8: EXAMPLE OF A PARTIALLY MULTIPLEXED ADDRESSING APPLICATION

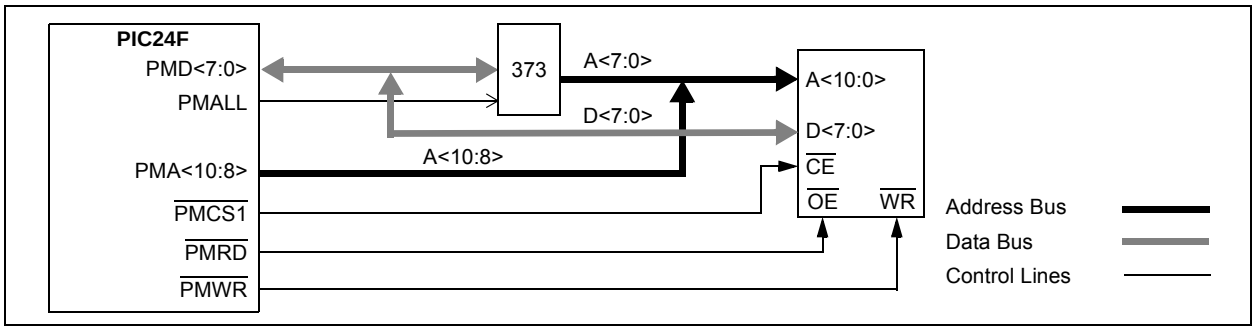


FIGURE 18-9: EXAMPLE OF AN 8-BIT MULTIPLEXED ADDRESS AND DATA APPLICATION

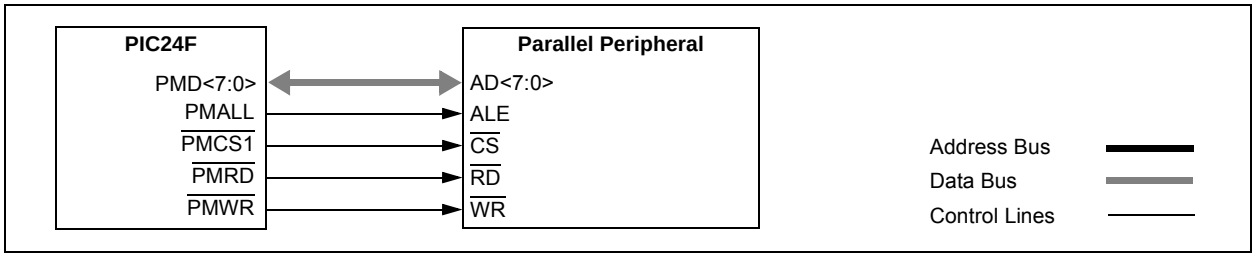


FIGURE 18-10: PARALLEL EEPROM EXAMPLE (UP TO 15-BIT ADDRESS, 8-BIT DATA)

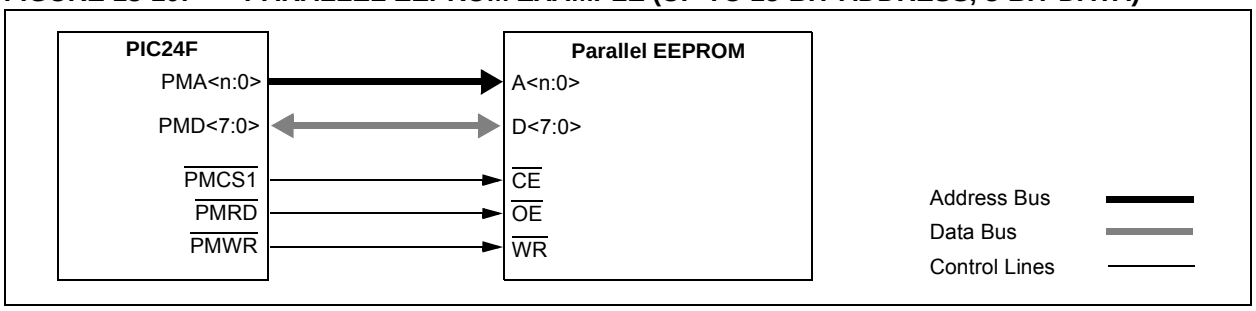


FIGURE 18-11: PARALLEL EEPROM EXAMPLE (UP TO 15-BIT ADDRESS, 16-BIT DATA)

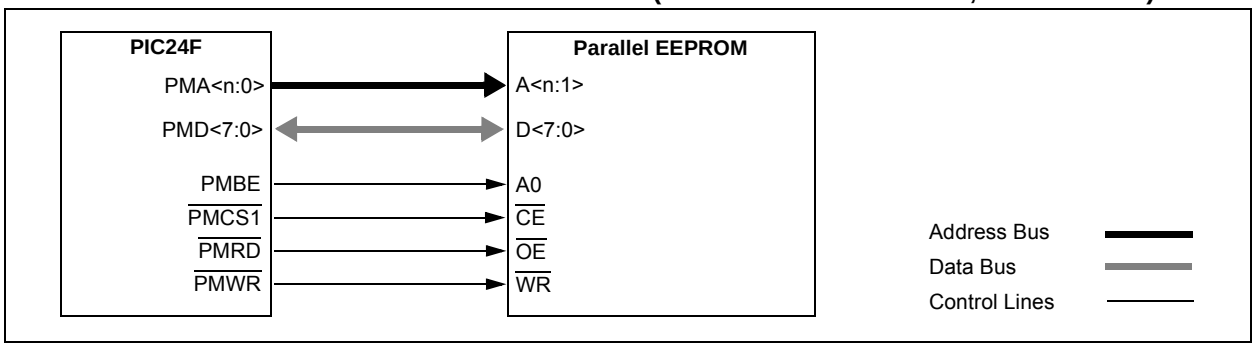
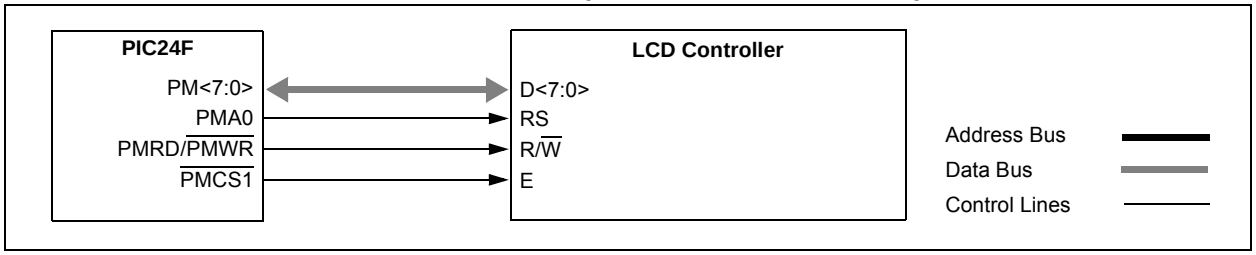


FIGURE 18-12: LCD CONTROL EXAMPLE (BYTE MODE OPERATION)



25.0 SPECIAL FEATURES

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the following sections of the *"PIC24F Family Reference Manual"*:

- **Section 9. "Watchdog Timer (WDT)"** (DS39697)
- **Section 32. "High-Level Device Integration"** (DS39719)
- **Section 33. "Programming and Diagnostics"** (DS39716)

PIC24FJ256GA110 family devices include several features intended to maximize application flexibility and reliability, and minimize cost through elimination of external components. These are:

- Flexible Configuration
- Watchdog Timer (WDT)
- Code Protection
- JTAG Boundary Scan Interface
- In-Circuit Serial Programming
- In-Circuit Emulation

25.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1'), to select various device configurations. These bits are mapped starting at program memory location F80000h. A detailed explanation of the various bit functions is provided in Register 25-1 through Register 25-5.

Note that address F80000h is beyond the user program memory space. In fact, it belongs to the configuration memory space (800000h-FFFFFFh) which can only be accessed using table reads and table writes.

25.1.1 CONSIDERATIONS FOR CONFIGURING PIC24FJ256GA110 FAMILY DEVICES

In PIC24FJ256GA110 family devices, the configuration bytes are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored in the three words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 25-1. These are packed representations of the actual device Configuration bits, whose actual locations are distributed among several locations in configuration space. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration registers during device Resets.

Note: Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Word for configuration data. This is to make certain that program code is not stored in this address when the code is compiled.

The upper byte of all Flash Configuration Words in program memory should always be '1111 1111'. This makes them appear to be NOP instructions in the remote event that their locations are ever executed by accident. Since Configuration bits are not implemented in the corresponding locations, writing '1's to these locations has no effect on device operation.

Note: Performing a page erase operation on the last page of program memory clears the Flash Configuration Words, enabling code protection as a result. Therefore, users should avoid performing page erase operations on the last page of program memory.

TABLE 25-1: FLASH CONFIGURATION WORD LOCATIONS FOR PIC24FJ256GA110 FAMILY DEVICES

Device	Configuration Word Addresses		
	1	2	3
PIC24FJ64GA1	ABFEh	ABFCh	ABFAh
PIC24FJ128GA1	157FEh	157FC	157FA
PIC24FJ192GA1	20BFEh	20BFC	20BFA
PIC24FJ256GA1	2ABFEh	2ABFC	2ABFA

REGISTER 25-2: CW2: FLASH CONFIGURATION WORD 2 (CONTINUED)

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Configuration bits
11 = Primary Oscillator disabled
10 = HS Oscillator mode selected
01 = XT Oscillator mode selected
00 = EC Oscillator mode selected

Note 1: Implemented in 100-pin devices only; otherwise unimplemented, read as '1'.

REGISTER 25-3: CW3: FLASH CONFIGURATION WORD 3

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
—	—	—	—	—	—	—	—
bit 23							bit 16

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
WPEND	WPCFG	WPDIS	—	—	—	—	—
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
WPFP7	WPFP6	WPFP5	WPFP4	WPFP3	WPFP2	WPFP1	WPFP0
bit 7							bit 0

Legend:		
R = Readable bit	PO = Program Once bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed	'1' = Bit is set	'0' = Bit is cleared

bit 23-16 **Reserved**

bit 15 **WPEND:** Segment Write Protection End Page Select bit
1 = Protected code segment upper boundary is at the last page of program memory; lower boundary is the code page specified by WPFP<7:0>
0 = Protected code segment lower boundary is at the bottom of program memory (000000h); upper boundary is the code page specified by WPFP<7:0>

bit 14 **WPCFG:** Configuration Word Code Page Protection Select bit
1 = Last page (at the top of program memory) and Flash Configuration Words are not protected if WPEND = 0
0 = Last page and Flash Configuration Words are code-protected if WPEND = 0

bit 13 **WPDIS:** Segment Write Protection Disable bit
1 = Segmented code protection disabled
0 = Segmented code protection enabled; protected segment defined by WPEND, WPCFG and WPFPx Configuration bits

bit 12-8 **Reserved**

bit 7-0 **WPFP<7:0>:** Protected Code Segment Boundary Page bits
Designates the 512-word program code page that is the boundary of the protected code segment, starting with Page 0 at the bottom of program memory.
If WPEND = 1:
First address of designated code page is the lower boundary of the segment.
If WPEND = 0:
Last address of designated code page is the upper boundary of the segment.

25.2.2 ON-CHIP REGULATOR AND POR

When the voltage regulator is enabled, it takes approximately 10 μs for it to generate output. During this time, designated as TVREG, code execution is disabled. TVREG is applied every time the device resumes operation after any power-down, including Sleep mode. The length of TVREG is determined by the PMSLP bit (RCON<8>), as described in **Section 25.2.5 "Voltage Regulator Standby Mode"**.

If the regulator is disabled, a separate Power-up Timer (PWRT) is automatically enabled. The PWRT adds a fixed delay of 64 ms nominal delay at device start-up (POR or BOR only). When waking up from Sleep with the regulator disabled, the PMSLP bit determines the wake-up time. When operating with the regulator disabled, setting PMSLP can decrease the device wake-up time.

25.2.3 ON-CHIP REGULATOR AND BOR

When the on-chip regulator is enabled, PIC24FJ256GA110 family devices also have a simple brown-out capability. If the voltage supplied to the regulator is inadequate to maintain the tracking level, the regulator Reset circuitry will generate a Brown-out Reset. This event is captured by the BOR flag bit (RCON<1>). The brown-out voltage specifications are provided in the *"PIC24FJ Family Reference Manual"*, **Section 7. "Reset"** (DS39712).

25.2.4 POWER-UP REQUIREMENTS

The on-chip regulator is designed to meet the power-up requirements for the device. If the application does not use the regulator, then strict power-up conditions must be adhered to. While powering up, VDDCORE must never exceed VDD by 0.3 volts.

Note:	For more information, see **Section 28.0 "Electrical Characteristics"**.

25.2.5 VOLTAGE REGULATOR STANDBY MODE

When enabled, the on-chip regulator always consumes a small incremental amount of current over IDD/IPD, including when the device is in Sleep mode, even though the core digital logic does not require power. To provide additional savings in applications where power resources are critical, the regulator automatically disables itself whenever the device goes into Sleep mode. This feature is controlled by the PMSLP bit (RCON<8>). By default, the bit is cleared, which removes power from the Flash program memory, and thus, enables Standby mode. When waking up from Standby mode, the regulator must wait for TVREG to expire before wake-up. This extra time is needed to ensure that the regulator can source enough current to power the Flash memory.

For applications which require a faster wake-up time, it is possible to disable regulator Standby mode. The PMSLP bit can be set to turn off Standby mode so that the Flash stays powered when in Sleep mode and the device can wake-up without waiting for TVREG. When PMSLP is set, the power consumption while in Sleep mode, will be approximately 40 μA higher than power consumption when the regulator is allowed to enter Standby mode.

25.3 Watchdog Timer (WDT)

For PIC24FJ256GA110 family devices, the WDT is driven by the LPRC Oscillator. When the WDT is enabled, the clock source is also enabled.

The nominal WDT clock source from LPRC is 31 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the FWPSA Configuration bit. With a 31 kHz input, the prescaler yields a nominal WDT time-out period (TWDT) of 1 ms in 5-bit mode or 4 ms in 7-bit mode.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPS<3:0> Configuration bits (CW1<3:0>), which allow the selection of a total of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler, time-out periods ranging from 1 ms to 131 seconds can be achieved.

The WDT, prescaler and postscaler are reset:

- On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSC bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a PWRSAV instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a CLRWDT instruction during normal execution

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake the device and code execution will continue from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE bits (RCON<3:2>) will need to be cleared in software after the device wakes up.

The WDT Flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

Note:	The CLRWDT and PWRSAV instructions clear the prescaler and postscaler counts when executed.

27.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers and dsPIC® digital signal controllers are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB C Compiler for Various Device Families
  - HI-TECH C for Various Device Families
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers
  - MPLAB ICD 3
  - PICkit™ 3 Debug Express
- Device Programmers
  - PICkit™ 2 Programmer
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

27.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - In-Circuit Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (C or assembly)
  - Mixed C and assembly
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

TABLE 28-5: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Conditions**		
Idle Current (IIDLE): Core Off, Clock On Base Current, PMD Bits are Set(2)						
DC40	220	310	µA	-40°C	2.0V(3)	1 MIPS
DC40a	220	310	µA	+25°C		
DC40b	220	310	µA	+85°C		
DC40c	260	350	µA	+125°C		
DC40d	300	390	µA	-40°C	3.3V(4)	
DC40e	300	390	µA	+25°C		
DC40f	320	420	µA	+85°C		
DC40g	340	450	µA	+125°C		
DC43	0.85	1.1	mA	-40°C	2.0V(3)	4 MIPS
DC43a	0.85	1.1	mA	+25°C		
DC43b	0.87	1.2	mA	+85°C		
DC43c	0.87	1.2	mA	+125°C		
DC43d	1.1	1.4	mA	-40°C	3.3V(4)	
DC43e	1.1	1.4	mA	+25°C		
DC43f	1.1	1.4	mA	+85°C		
DC43g	1.1	1.5	mA	+125°C		
DC47	4.4	5.6	mA	-40°C	2.5V(3)	16 MIPS
DC47a	4.4	5.6	mA	+25°C		
DC47b	4.4	5.6	mA	+85°C		
DC47c	4.4	5.6	mA	+125°C		
DC47d	4.4	5.6	mA	-40°C	3.3V(4)	
DC47e	4.4	5.6	mA	+25°C		
DC47f	4.4	5.6	mA	+85°C		
DC47g	4.4	5.6	mA	+125°C		
DC50	1.1	1.4	mA	-40°C	2.0V(3)	FRC (4 MIPS)
DC50a	1.1	1.4	mA	+25°C		
DC50b	1.1	1.4	mA	+85°C		
DC50c	1.2	1.5	mA	+125°C		
DC50d	1.4	1.8	mA	-40°C	3.3V(4)	
DC50e	1.4	1.8	mA	+25°C		
DC50f	1.4	1.8	mA	+85°C		
DC50g	1.4	1.8	mA	+125°C		

Note 1: Data in "Typical" column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Base IIDLE current is measured with core off, clock on, all modules off and all of the Peripheral Module Disable (PMD) bits are set.

3: On-chip voltage regulator disabled (ENVREG tied to VSS).

4: On-chip voltage regulator enabled (ENVREG tied to VDD).

FIGURE 28-18: I²C™ BUS DATA TIMING CHARACTERISTICS (MASTER MODE)

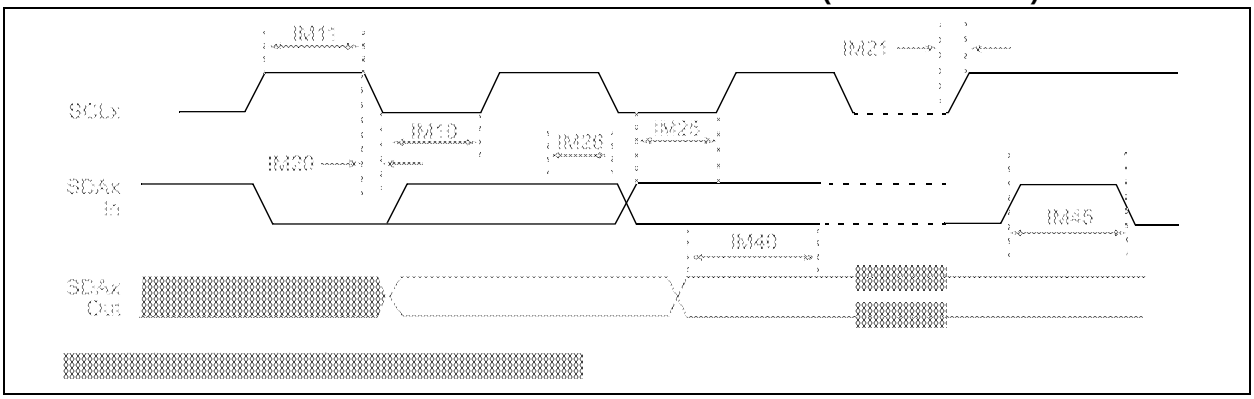


TABLE 28-31: I²C™ BUS DATA TIMING REQUIREMENTS (MASTER MODE)

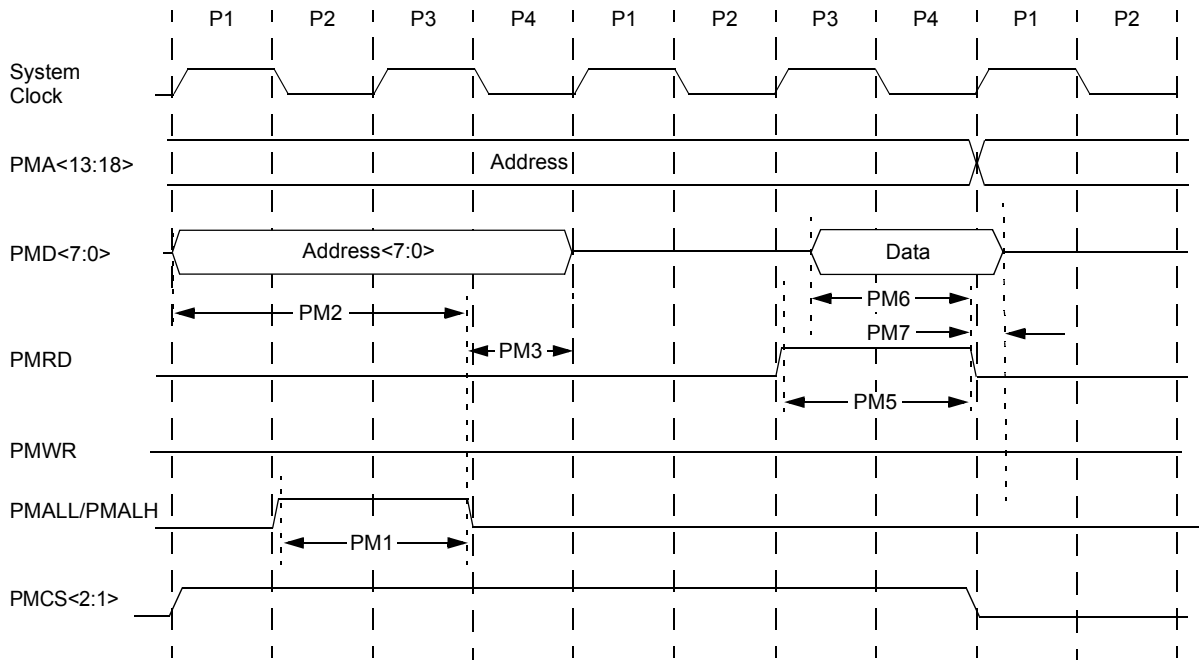
AC CHARACTERISTICS				Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C (Industrial)			
Param No.	**Symbol**	**Characteristic**		**Min(1)**	**Max**	**Units**	**Conditions**
IM10	TLO:SCL	Clock Low Time	100 kHz mode	TCY/2 (BRG + 1)	—	µs	—
			400 kHz mode	TCY/2 (BRG + 1)	—	µs	—
			1 MHz mode(2)	TCY/2 (BRG + 1)	—	µs	—
IM11	THI:SCL	Clock High Time	100 kHz mode	TCY/2 (BRG + 1)	—	µs	—
			400 kHz mode	TCY/2 (BRG + 1)	—	µs	—
			1 MHz mode(2)	TCY/2 (BRG + 1)	—	µs	—
IM20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	100	ns	
IM21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	300	ns	
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode(2)	TBD	—	ns	
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	ns	—
			400 kHz mode	0	0.9	µs	
			1 MHz mode(2)	TBD	—	ns	
IM40	TAA:SCL	Output Valid From Clock	100 kHz mode	—	3500	ns	—
			400 kHz mode	—	1000	ns	—
			1 MHz mode(2)	—	—	ns	—
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	µs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
			1 MHz mode(2)	TBD	—	µs	
IM50	CB	Bus Capacitive Loading		—	400	pF	—

Legend: TBD = To Be Determined

Note 1: BRG is the value of the I²C Baud Rate Generator. Refer to **Section 16.3 "Setting Baud Rate When Operating as a Bus Master"** for details.

2: Maximum pin capacitance = 10 pF for all I²C pins (for 1 MHz mode only).

FIGURE 28-22: PARALLEL MASTER PORT READ TIMING DIAGRAM



Operating Conditions: 2.0V < VCC < 3.6V, -40°C < TA < +85°C unless otherwise stated.

TABLE 28-35: PARALLEL MASTER PORT READ TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param. No	**Symbol**	**Characteristics(1)**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
PM1		PMALL/PMALH Pulse Width	—	0.5 TCY	—	ns	
PM2		Address Out Valid to PMALL/PMALH Invalid (address setup time)(2)	—	0.75 TCY	—	ns	
PM3		PMALL/PMALH Invalid to Address Out Invalid (address hold time)	—	0.25 TCY	—	ns	
PM5		PMRD Pulse Width	—	0.5 TCY	—	ns	
PM6		Data In to PMRD or PMENB Inactive state	150	—	—	ns	
PM7		PMRD or PMENB Inactive to Data In Invalid (data hold time)	—	—	5	ns	

Note 1: Wait states disabled for all cases.

2: The setup time for the LSB and the MSB of the address are not the same; the setup time for the LSB is 0.5 TCY and for the MSB is 0.75 TCY.

Output Compare
Cascaded (32-Bit) Mode 167
Operations 168
Synchronous and Trigger Modes 167
Output Compare with Dedicated Timer 167

P

Packaging 305
Details 307
Marking 305
Parallel Master Port. *See* PMP. 201
Peripheral Enable Bits 126
Peripheral Module Disable Bits 126
Peripheral Pin Select (PPS) 129
Alternate Fixed Pin Mapping 130
Available Peripherals and Pins 130
Configuration Control 133
Considerations for Use 134
Input Mapping 130
Mapping Exceptions 133
Output Mapping 130
Peripheral Priority 130
Pinout Descriptions 15–22
PMSLP Bit 66
and Wake-up Time 125
PMSLP bit
and Wake-up Time 252
Power-Saving Features 125
Modes
Doze 126
Idle 126
Sleep 125
Power-up Requirements 252
Product Identification System 328
Program Memory
Access Using Table Instructions 55
Address Space 35
Addressing 53
Flash Configuration Words 36
Memory Maps 35
Organization 36
Program Space Visibility 56
Program Space Visibility (PSV) 56
Program Verification 253
Programmer's Model 29
Pulse-Width Modulation (PWM) Mode 169
Pulse-Width Modulation. *See* PWM.
PWM
Duty Cycle and Period 170

R

Reader Response 327
Register Maps
ADC 49
Comparators 50
CPU Core 39
CRC 50
CTMU 49
I^2C 45
ICN 40
Input Capture 43
Interrupt Controller 41
NVM 52
Output Compare 44
Pad Configuration 48
Parallel Master/Slave Port 50
Peripheral Pin Select 51
PMD 52
PORTA 47
PORTB 47
PORTC 47
PORTD 47
PORTE 48
PORTF 48
Real-Time Clock and Calendar 50
SPI 46
System 52
Timers 42
UART 46
Registers
AD1CHS (A/D Input Select) 230
AD1CON1 (A/D Control 1) 227
AD1CON2 (A/D Control 2) 228
AD1CON3 (A/D Control 3) 229
AD1CSSL (A/D Input Scan Select Low) 232
AD1PCFGH (A/D Port Configuration High) 231
AD1PCFGL (A/D Port Configuration Low) 231
ALCFGRPT (Alarm Configuration) 215
ALMINSEC (Alarm Minutes and Seconds Value) 219
ALMTHDY (Alarm Month and Day Value) 218
ALTRP (Alternate Peripheral Pin Mapping) 154
ALWDHR (Alarm Weekday and Hours Value) 218
CLKDIV (Clock Divider) 119
CMSTAT (Comparator Status) 238
CMxCON (Comparator x Control) 237
CORCON (CPU Control) 33, 75
CRCCON (CRC Control) 223
CRCXOR (CRC XOR Polynomial) 224
CTMUCON (CTMU Control) 243
CTMUICON (CTMU Current Control) 244
CVRCON (Comparator Voltage Reference Control) 240
CW1 (Flash Configuration Word 1) 246
CW2 (Flash Configuration Word 2) 248
CW3 (Flash Configuration Word 3) 249
DEVID (Device ID) 250
DEVREV (Device Revision) 250
I2CxCON (I2Cx Control) 188
I2CxMSK (I2Cx Slave Mode Address Mask) 192
I2CxSTAT (I2Cx Status) 190
ICxCON1 (Input Capture x Control 1) 165
ICxCON2 (Input Capture x Control 2) 166
IEC0 (Interrupt Enable Control 0) 84
IEC1 (Interrupt Enable Control 1) 85
IEC2 (Interrupt Enable Control 2) 87
IEC3 (Interrupt Enable Control 3) 88
IEC4 (Interrupt Enable Control 4) 89
IEC5 (Interrupt Enable Control 5) 90
IFS0 (Interrupt Flag Status 0) 78
IFS1 (Interrupt Flag Status 1) 79
IFS2 (Interrupt Flag Status 2) 80
IFS3 (Interrupt Flag Status 3) 81
IFS4 (Interrupt Flag Status 4) 82
IFS5 (Interrupt Flag Status 5) 83
INTCON1 (Interrupt Control 1) 76
INTCON2 (Interrupt Control 2) 77
INTTREG (Interrupt Control and Status) 112
IPC0 (Interrupt Priority Control 0) 91
IPC1 (Interrupt Priority Control 1) 92
IPC10 (Interrupt Priority Control 10) 101
IPC11 (Interrupt Priority Control 11) 102