



Welcome to [E-XFL.COM](https://www.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	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I ² C, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	48KB (16K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 13x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj48ga004-e-pt

2.6 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 8.0 “Oscillator Configuration”** for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

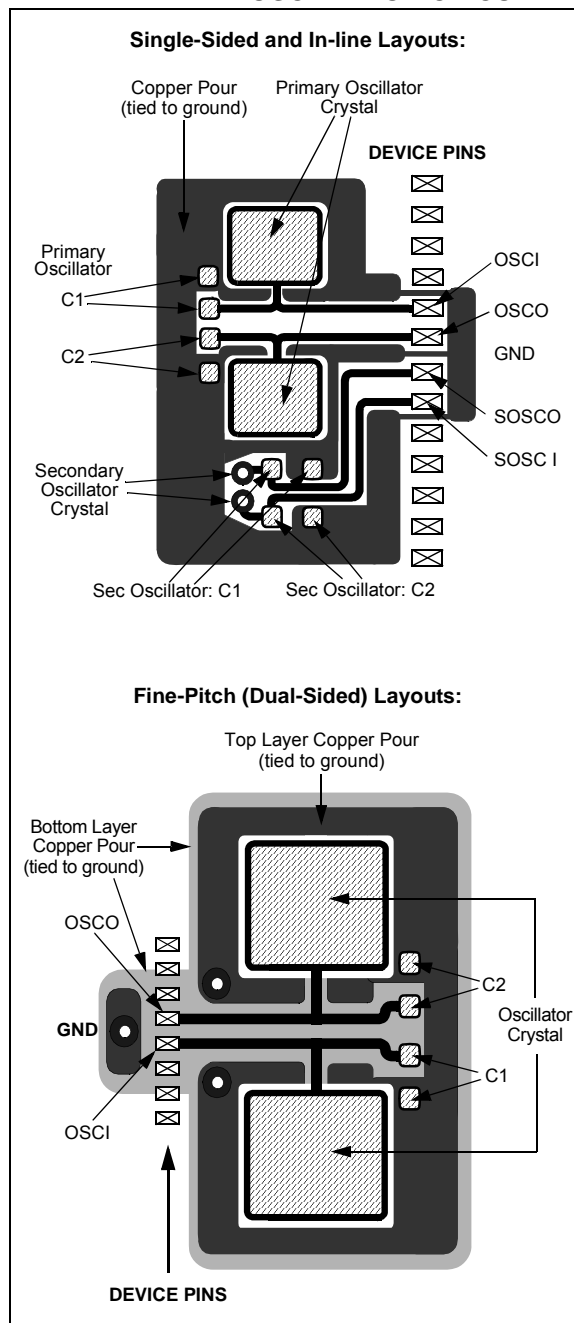
Layout suggestions are shown in Figure 2-5. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times and other similar noise).

For additional information and design guidance on oscillator circuits, please refer to these Microchip Application Notes, available at the corporate web site (www.microchip.com):

- AN826, “Crystal Oscillator Basics and Crystal Selection for rPIC™ and PICmicro® Devices”
- AN849, “Basic PICmicro® Oscillator Design”
- AN943, “Practical PICmicro® Oscillator Analysis and Design”
- AN949, “Making Your Oscillator Work”

FIGURE 2-5: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



PIC24FJ64GA004 FAMILY

2.7 Configuration of Analog and Digital Pins During ICSP Operations

If an ICSP compliant emulator is selected as a debugger, it automatically initializes all of the A/D input pins (ANx) as “digital” pins. Depending on the particular device, this is done by setting all bits in the ADnPCFG register(s), or clearing all bit in the ANSx registers.

All PIC24F devices will have either one or more ADnPCFG registers or several ANSx registers (one for each port); no device will have both. Refer to **Section 21.0 “10-Bit High-Speed A/D Converter”** for more specific information.

The bits in these registers that correspond to the A/D pins that initialized the emulator must not be changed by the user application firmware; otherwise, communication errors will result between the debugger and the device.

If your application needs to use certain A/D pins as analog input pins during the debug session, the user application must modify the appropriate bits during initialization of the A/D module, as follows:

- For devices with an ADnPCFG register, clear the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.
- For devices with ANSx registers, set the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.

When a Microchip debugger/emulator is used as a programmer, the user application firmware must correctly configure the ADnPCFG or ANSx registers. Automatic initialization of this register is only done during debugger operation. Failure to correctly configure the register(s) will result in all A/D pins being recognized as analog input pins, resulting in the port value being read as a logic ‘0’, which may affect user application functionality.

2.8 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 k Ω to 10 k Ω resistor to Vss on unused pins and drive the output to logic low.

PIC24FJ64GA004 FAMILY

REGISTER 7-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPF1IF	T3IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
T2IF	OC2IF	IC2IF	—	T1IF	OC1IF	IC1IF	INT0IF
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 **AD1IF:** A/D Conversion Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 12 **U1TXIF:** UART1 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 11 **U1RXIF:** UART1 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 10 **SPI1IF:** SPI1 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 9 **SPF1IF:** SPI1 Fault Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 8 **T3IF:** Timer3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 7 **T2IF:** Timer2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 6 **OC2IF:** Output Compare Channel 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 5 **IC2IF:** Input Capture Channel 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 4 **Unimplemented:** Read as '0'
- bit 3 **T1IF:** Timer1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred
- bit 2 **OC1IF:** Output Compare Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

PIC24FJ64GA004 FAMILY

8.0 OSCILLATOR CONFIGURATION

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”, “Oscillator” (DS39700).

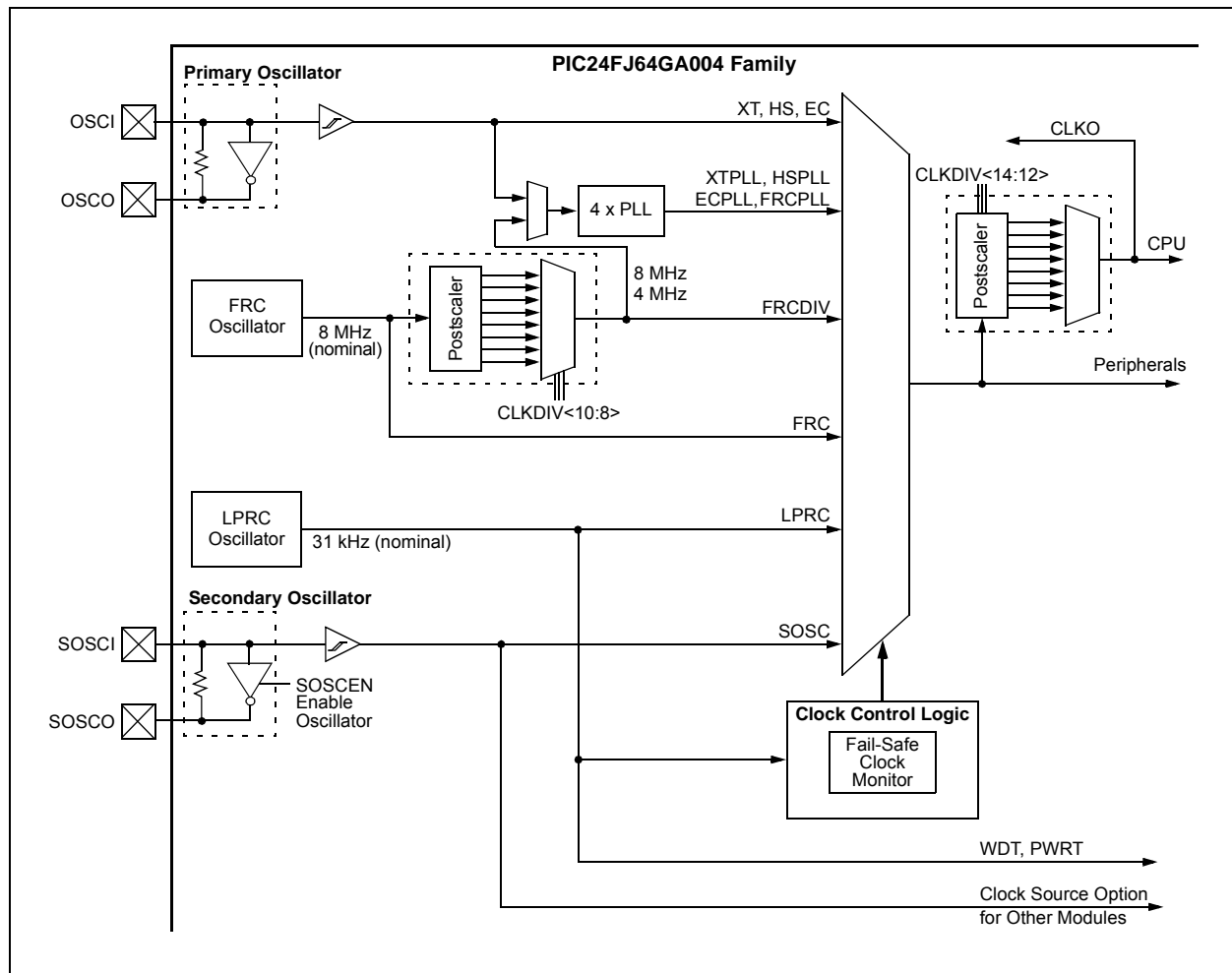
- Software-controllable switching between various clock sources
- Software-controllable postscaler for selective clocking of CPU for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown

A simplified diagram of the oscillator system is shown in Figure 8-1.

The oscillator system for PIC24FJ64GA004 family devices has the following features:

- A total of four external and internal oscillator options as clock sources, providing 11 different clock modes
- On-chip 4x PLL to boost internal operating frequency on select internal and external oscillator sources

FIGURE 8-1: PIC24FJ64GA004 FAMILY CLOCK DIAGRAM



PIC24FJ64GA004 FAMILY

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 Flash Configuration Word 2 must be programmed to '0'. (Refer to **Section 24.1 "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.

10.4.3.2 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 two 5-bit fields; each field being associated with one RPN pin (see Register 10-15 through Register 10-27). The value of the bit field corresponds to one of the peripherals and that peripheral's output is mapped to the pin (see Table 10-3).

Because of the mapping technique, the list of peripherals for output mapping also includes a null value of '00000'. This permits any given pin to remain disconnected from the output of any of the pin-selectable peripherals.

TABLE 10-3: SELECTABLE OUTPUT SOURCES (MAPS FUNCTION TO OUTPUT)

Function	Output Function Number ⁽¹⁾	Output Name
NULL ⁽²⁾	0	NULL
C1OUT	1	Comparator 1 Output
C2OUT	2	Comparator 2 Output
U1TX	3	UART1 Transmit
U1RTS ⁽³⁾	4	UART1 Request-to-Send
U2TX	5	UART2 Transmit
U2RTS ⁽³⁾	6	UART2 Request-to-Send
SDO1	7	SPI1 Data Output
SCK1OUT	8	SPI1 Clock Output
SS1OUT	9	SPI1 Slave Select Output
SDO2	10	SPI2 Data Output
SCK2OUT	11	SPI2 Clock Output
SS2OUT	12	SPI2 Slave Select Output
OC1	18	Output Compare 1
OC2	19	Output Compare 2
OC3	20	Output Compare 3
OC4	21	Output Compare 4
OC5	22	Output Compare 5

- Note 1:** Value assigned to the RPN<4:0> pins corresponds to the peripheral output function number.
- 2:** The NULL function is assigned to all RPN outputs at device Reset and disables the RPN output function.
- 3:** IrDA[®] BCLK functionality uses this output.

10.4.3.3 Mapping Limitations

The control schema of the Peripheral Pin Select is extremely flexible. Other than systematic blocks that prevent signal contention, caused by two physical pins being configured as the same functional input or two functional outputs configured as the same pin, there are no hardware enforced lockouts. The flexibility extends to the point of allowing a single input to drive multiple peripherals or a single functional output to drive multiple output pins.

10.4.4 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC24F devices include three features to prevent alterations to the peripheral map:

- Control register lock sequence
- Continuous state monitoring
- Configuration bit remapping lock

10.4.4.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes will appear to execute normally, but the contents of the registers will remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear IOLOCK, a specific command sequence must be executed:

1. Write 46h to OSCCON<7:0>.
2. Write 57h to OSCCON<7:0>.
3. Clear (or set) IOLOCK as a single operation.

Unlike the similar sequence with the oscillator's LOCK bit, IOLOCK remains in one state until changed. This allows all of the Peripheral Pin Selects to be configured with a single unlock sequence, followed by an update to all control registers, then locked with a second lock sequence.

10.4.4.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a Configuration Mismatch Reset will be triggered.

PIC24FJ64GA004 FAMILY

10.4.4.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY (CW2<4>) Configuration bit blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure will not execute and the Peripheral Pin Select Control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session. Programming IOL1WAY allows users unlimited access (with the proper use of the unlock sequence) to the Peripheral Pin Select registers.

10.4.5 CONSIDERATIONS FOR PERIPHERAL PIN SELECTION

The ability to control Peripheral Pin Selection introduces several considerations into application design that could be overlooked. This is particularly true for several common peripherals that are available only as remappable peripherals.

The main consideration is that the Peripheral Pin Selects are not available on default pins in the device's default (Reset) state. Since all RPINRx registers reset to '11111' and all RPORx registers reset to '00000', all Peripheral Pin Select inputs are tied to RP31 and all Peripheral Pin Select outputs are disconnected.

Note: In tying Peripheral Pin Select inputs to RP31, RP31 does not have to exist on a device for the registers to be reset to it.

This situation requires the user to initialize the device with the proper peripheral configuration before any other application code is executed. Since the IOLOCK bit resets in the unlocked state, it is not necessary to execute the unlock sequence after the device has come out of Reset. For application safety, however, it is best to set IOLOCK and lock the configuration after writing to the control registers.

Because the unlock sequence is timing critical, it must be executed as an assembly language routine in the same manner as changes to the oscillator configuration. If the bulk of the application is written in C or another high-level language, the unlock sequence should be performed by writing in-line assembly.

Choosing the configuration requires the review of all Peripheral Pin Selects and their pin assignments, especially those that will not be used in the application. In all cases, unused pin-selectable peripherals should be disabled completely. Unused peripherals should have their inputs assigned to an unused RPn pin function. I/O pins with unused RPn functions should be configured with the null peripheral output.

The assignment of a peripheral to a particular pin does not automatically perform any other configuration of the pin's I/O circuitry. In theory, this means adding a pin-selectable output to a pin may mean inadvertently driving an existing peripheral input when the output is driven. Users must be familiar with the behavior of other fixed peripherals that share a remappable pin and know when to enable or disable them. To be safe, fixed digital peripherals that share the same pin should be disabled when not in use.

Along these lines, configuring a remappable pin for a specific peripheral does not automatically turn that feature on. The peripheral must be specifically configured for operation and enabled, as if it were tied to a fixed pin. Where this happens in the application code (immediately following device Reset and peripheral configuration or inside the main application routine) depends on the peripheral and its use in the application.

A final consideration is that Peripheral Pin Select functions neither override analog inputs, nor reconfigure pins with analog functions for digital I/O. If a pin is configured as an analog input on device Reset, it must be explicitly reconfigured as a digital I/O when used with a Peripheral Pin Select.

Example 10-2 shows a configuration for bidirectional communication with flow control using UART1. The following input and output functions are used:

- Input Functions: U1RX, U1CTS
- Output Functions: U1TX, U1RTS

EXAMPLE 10-2: CONFIGURING UART1 INPUT AND OUTPUT FUNCTIONS

```
// Unlock Registers
__builtin_write_OSCCONL(OSCCON & 0xBF);

// Configure Input Functions (Table 10-2)
// Assign U1RX To Pin RP0
RPINR18bits.U1RXR = 0;
// Assign U1CTS To Pin RP1
RPINR18bits.U1CTSR = 1;
// Configure Output Functions (Table 10-3)
// Assign U1TX To Pin RP2
RPOR1bits.RP2R = 3;
// Assign U1RTS To Pin RP3
RPOR1bits.RP3R = 4;

// Lock Registers
__builtin_write_OSCCONL(OSCCON | 0x40);
```

PIC24FJ64GA004 FAMILY

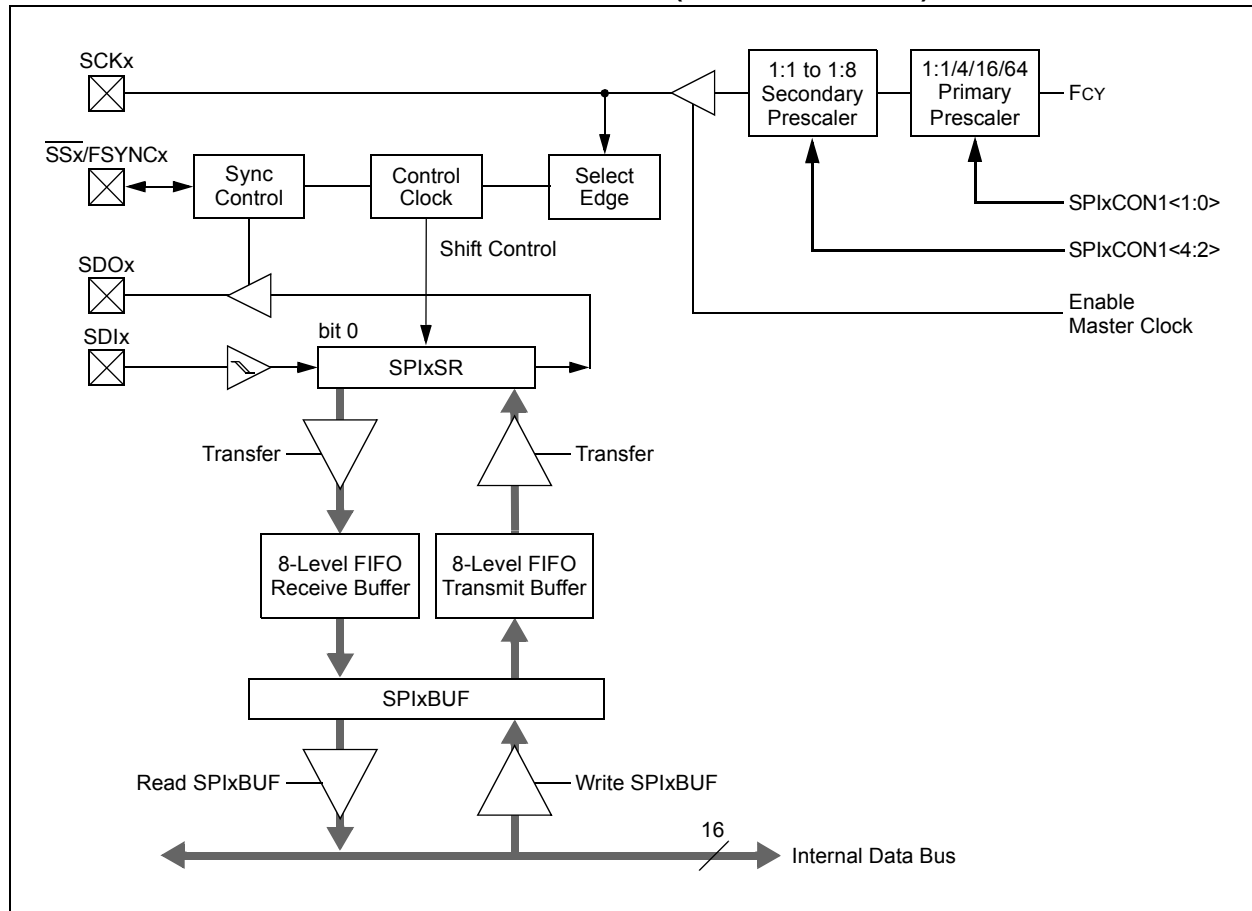
To set up the SPIx module for the Enhanced Buffer Master mode of operation:

- If using interrupts:
 - Clear the SPIxIF bit in the respective IFSx register.
 - Set the SPIxIE bit in the respective IECx register.
 - Write the SPIxIP bits in the respective IPCx register.
- Write the desired settings to the SPIxCON1 and SPIxCON2 registers with the MSTEN bit (SPIxCON1<5>) = 1.
- Clear the SPIROV bit (SPIxSTAT<6>).
- Select Enhanced Buffer mode by setting the SPIBEN bit (SPIxCON2<0>).
- Enable SPIx operation by setting the SPIEN bit (SPIxSTAT<15>).
- Write the data to be transmitted to the SPIxBUF register. Transmission (and reception) will start as soon as data is written to the SPIxBUF register.

To set up the SPIx module for the Enhanced Buffer Slave mode of operation:

- Clear the SPIxBUF register.
- If using interrupts:
 - Clear the SPIxIF bit in the respective IFSx register.
 - Set the SPIxIE bit in the respective IECx register.
 - Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
- Write the desired settings to the SPIxCON1 and SPIxCON2 registers with the MSTEN bit (SPIxCON1<5>) = 0.
- Clear the SMP bit.
- If the CKE bit is set, then the SSx pin must be set, thus enabling the SSx pin.
- Clear the SPIROV bit (SPIxSTAT<6>).
- Select Enhanced Buffer mode by setting the SPIBEN bit (SPIxCON2<0>).
- Enable SPIx operation by setting the SPIEN bit (SPIxSTAT<15>).

FIGURE 15-2: SPIx MODULE BLOCK DIAGRAM (ENHANCED MODE)



--



PIC24FJ64GA004 FAMILY

REGISTER 18-3: PMADDR: PARALLEL PORT ADDRESS REGISTER

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	CS1	—	—	—	ADDR10 ⁽¹⁾	ADDR9 ⁽¹⁾	ADDR8 ⁽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
ADDR7 ⁽¹⁾	ADDR6 ⁽¹⁾	ADDR5 ⁽¹⁾	ADDR4 ⁽¹⁾	ADDR3 ⁽¹⁾	ADDR2 ⁽¹⁾	ADDR1 ⁽¹⁾	ADDR0 ⁽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 **Unimplemented:** Read as '0'

bit 14 **CS1:** Chip Select 1 bit

1 = Chip Select 1 is active

0 = Chip Select 1 is inactive

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

bit 10-0 **ADDR<10:0>:** Parallel Port Destination Address bits⁽¹⁾

Note 1: PMA<10:2> bits are not available on 28-pin devices.

REGISTER 18-4: PMAEN: PARALLEL PORT ENABLE REGISTER

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	PTEN14	—	—	—	PTEN10 ⁽¹⁾	PTEN9 ⁽¹⁾	PTEN8 ⁽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
PTEN7 ⁽¹⁾	PTEN6 ⁽¹⁾	PTEN5 ⁽¹⁾	PTEN4 ⁽¹⁾	PTEN3 ⁽¹⁾	PTEN2 ⁽¹⁾	PTEN1	PTEN0
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 **PTEN14:** PMCS1 Strobe Enable bit

1 = PMCS1 pin functions as chip select

0 = PMCS1 pin functions as port I/O

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

bit 10-2 **PTEN<10:2>:** PMP Address Port Enable bits⁽¹⁾

1 = PMA<10:2> function as PMP address lines

0 = PMA<10:2> function as port I/O

bit 1-0 **PTEN<1:0>:** PMALH/PMALL Strobe Enable bits

1 = PMA1 and PMA0 function as either PMA<1:0> or PMALH and PMALL

0 = PMA1 and PMA0 pads functions as port I/O

Note 1: PMA<10:2> bits are not available on 28-pin devices.

PIC24FJ64GA004 FAMILY

REGISTER 18-5: PMSTAT: PARALLEL PORT STATUS REGISTER

R-0	R/W-0, HS	U-0	U-0	R-0	R-0	R-0	R-0
IBF	IBOV	—	—	IB3F	IB2F	IB1F	IB0F
bit 15				bit 8			

R-1	R/W-0, HS	U-0	U-0	R-1	R-1	R-1	R-1
OBE	OBUE	—	—	OB3E	OB2E	OB1E	OB0E
bit 7				bit 0			

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

- bit 15 **IBF:** Input Buffer Full Status bit
1 = All writable Input Buffer registers are full
0 = Some or all of the writable Input Buffer registers are empty
- bit 14 **IBOV:** Input Buffer Overflow Status bit
1 = A write attempt to a full Input Byte register occurred (must be cleared in software)
0 = No overflow occurred
- bit 13-12 **Unimplemented:** Read as '0'
- bit 11-8 **IB3F:IB0F:** Input Buffer x Status Full bits
1 = Input Buffer x contains data that has not been read (reading buffer will clear this bit)
0 = Input Buffer x does not contain any unread data
- bit 7 **OBE:** Output Buffer Empty Status bit
1 = All readable Output Buffer registers are empty
0 = Some or all of the readable Output Buffer registers are full
- bit 6 **OBUE:** Output Buffer Underflow Status bit
1 = A read occurred from an empty Output Byte register (must be cleared in software)
0 = No underflow occurred
- bit 5-4 **Unimplemented:** Read as '0'
- bit 3-0 **OB3E:OB0E** Output Buffer x Status Empty bits
1 = Output Buffer x is empty (writing data to the buffer will clear this bit)
0 = Output Buffer x contains data that has not been transmitted

PIC24FJ64GA004 FAMILY

REGISTER 19-3: ALCFGRPT: ALARM CONFIGURATION 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
ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0
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
ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0
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 **ALRMEN:** Alarm Enable bit
 1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 00h and CHIME = 0)
 0 = Alarm is disabled
- bit 14 **CHIME:** Chime Enable bit
 1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 00h to FFh
 0 = Chime is disabled; ARPT<7:0> bits stop once they reach 00h
- bit 13-10 **AMASK<3:0>:** Alarm Mask Configuration bits
 0000 = Every half second
 0001 = Every second
 0010 = Every 10 seconds
 0011 = Every minute
 0100 = Every 10 minutes
 0101 = Every hour
 0110 = Once a day
 0111 = Once a week
 1000 = Once a month
 1001 = Once a year (except when configured for February 29th, once every 4 years)
 101x = Reserved; do not use
 11xx = Reserved; do not use
- bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits
 Points to the corresponding Alarm Value registers when reading ALRMVALH and ALRMVALL registers; the ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.
ALRMVAL<15:8>:
 00 = ALRMMIN
 01 = ALRMWD
 10 = ALRMMNTH
 11 = Unimplemented
ALRMVAL<7:0>:
 00 = ALRMSEC
 01 = ALRMHR
 10 = ALRMDAY
 11 = Unimplemented
- bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits
 11111111 = Alarm will repeat 255 more times
 ...
 00000000 = Alarm will not repeat
 The counter decrements on any alarm event. The counter is prevented from rolling over from 00h to FFh unless CHIME = 1.

PIC24FJ64GA004 FAMILY

FIGURE 19-2: ALARM MASK SETTINGS

Alarm Mask Setting (AMASK<3:0>)	Day of the Week	Month	Day	Hours	Minutes	Seconds
0000 – Every half second 0001 – Every second	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>
0010 – Every 10 seconds	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> s
0011 – Every minute	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> s
0100 – Every 10 minutes	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> m	<input type="checkbox"/> <input type="checkbox"/> s
0101 – Every hour	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> m	<input type="checkbox"/> <input type="checkbox"/> s
0110 – Every day	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> h	<input type="checkbox"/> h	<input type="checkbox"/> <input type="checkbox"/> s
0111 – Every week	<input type="checkbox"/> d	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> h	<input type="checkbox"/> h	<input type="checkbox"/> <input type="checkbox"/> s
1000 – Every month	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> d	<input type="checkbox"/> h	<input type="checkbox"/> h	<input type="checkbox"/> <input type="checkbox"/> s
1001 – Every year ⁽¹⁾	<input type="checkbox"/>	<input type="checkbox"/> m	<input type="checkbox"/> m	<input type="checkbox"/> d	<input type="checkbox"/> d	<input type="checkbox"/> <input type="checkbox"/> s

Note 1: Annually, except when configured for February 29.

PIC24FJ64GA004 FAMILY

REGISTER 21-1: AD1CON1: A/D CONTROL REGISTER 1

R/W-0	U-0	R/C-0	U-0	U-0	U-0	R/W-0	R/W-0
ADON ⁽¹⁾	—	ADSIDL	—	—	—	FORM1	FORM0
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0, HSC	R/W-0, HSC
SSRC2	SSRC1	SSRC0	—	—	ASAM	SAMP	DONE
bit 7						bit 0	

Legend:	C = Clearable 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 **ADON:** A/D Operating Mode bit⁽¹⁾
1 = A/D Converter module is operating
0 = A/D Converter is off
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **ADSIDL:** A/D Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode
- bit 12-10 **Unimplemented:** Read as '0'
- bit 9-8 **FORM<1:0>:** Data Output Format bits
11 = Signed fractional (sddd dddd dd00 0000)
10 = Fractional (dddd dddd dd00 0000)
01 = Signed integer (ssss sssd dddd dddd)
00 = Integer (0000 00dd dddd dddd)
- bit 7-5 **SSRC<2:0>:** Conversion Trigger Source Select bits
111 = Internal counter ends sampling and starts conversion (auto-convert)
110 = Reserved
10x = Reserved
011 = Reserved
010 = Timer3 compare ends sampling and starts conversion
001 = Active transition on INT0 pin ends sampling and starts conversion
000 = Clearing the SAMP bit ends sampling and starts conversion
- bit 4-3 **Unimplemented:** Read as '0'
- bit 2 **ASAM:** A/D Sample Auto-Start bit
1 = Sampling begins immediately after last conversion completes; SAMP bit is auto-set
0 = Sampling begins when SAMP bit is set
- bit 1 **SAMP:** A/D Sample Enable bit
1 = A/D Sample-and-Hold (S/H) amplifier is sampling input
0 = A/D Sample-and-Hold amplifier is holding
- bit 0 **DONE:** A/D Conversion Status bit
1 = A/D conversion is done
0 = A/D conversion is NOT done

Note 1: The ADC1BUFn registers do not retain their values when ADON is cleared. Read out any conversion values from the buffer before disabling the module.

PIC24FJ64GA004 FAMILY

24.0 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”:

- “**Watchdog Timer (WDT)**” (DS39697)
- “**High-Level Device Integration**” (DS39719)
- “**Programming and Diagnostics**” (DS39716)

PIC24FJ64GA004 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

24.1 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 complete list of locations is shown in Table 24-1. A detailed explanation of the various bit functions is provided in Register 24-1 through Register 24-4.

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.

24.1.1 CONSIDERATIONS FOR CONFIGURING PIC24FJ64GA004 FAMILY DEVICES

In PIC24FJ64GA004 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 two words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 24-1. These are packed representations of the actual device Configuration bits, whose actual locations are distributed among five 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.

TABLE 24-1: FLASH CONFIGURATION WORD LOCATIONS FOR PIC24FJ64GA004 FAMILY DEVICES

Device	Configuration Word Addresses	
	1	2
PIC24FJ16GA	002BFEh	002BFCh
PIC24FJ32GA	0057FEh	0057FCh
PIC24FJ48GA	0083FEh	0083FCh
PIC24FJ64GA	00ABFEh	00ABFCh

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 Configuration bits are reloaded from the Flash Configuration Word on any device Reset.

The upper byte of both 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.

PIC24FJ64GA004 FAMILY

REGISTER 24-2: CW2: FLASH CONFIGURATION WORD 2 (CONTINUED)

- bit 4 **IOL1WAY:** IOLOCK One-Way Set Enable bit
 1 = The IOLOCK (OSCCON<6>) bit can be set once, provided the unlock sequence has been completed. Once set, the Peripheral Pin Select registers cannot be written to a second time.
 0 = The IOLOCK (OSCCON<6>) bit can be set and cleared as needed, provided the unlock sequence has been completed
- bit 3 **Reserved**
- bit 2 **I2C1SEL:** I2C1 Pin Select bit
 1 = Use default SCL1/SDA1 pins
 0 = Use alternate SCL1/SDA1 pins
- bit 1-0 **POSCMD<1:0>:** Primary Oscillator Configuration bits
 11 = Primary oscillator is disabled
 10 = HS Oscillator mode is selected
 01 = XT Oscillator mode is selected
 00 = EC Oscillator mode is selected

Note 1: These bits are implemented only in devices with a major silicon revision level of B or later (DEVREV register value is 3042h or greater). Refer to **Section 28.0 “Packaging Information”** in the device data sheet for the location and interpretation of product date codes.

REGISTER 24-3: DEVID: DEVICE ID REGISTER

U	U	U	U	U	U	U	U
—	—	—	—	—	—	—	—
bit 23							bit 16

U	U	R	R	R	R	R	R
—	—	FAMID7	FAMID6	FAMID5	FAMID4	FAMID3	FAMID2
bit 15							bit 8

R	R	R	R	R	R	R	R
FAMID1	FAMID0	DEV5	DEV4	DEV3	DEV2	DEV1	DEV0
bit 7							bit 0

Legend: R = Read-only bit U = Unimplemented bit

- bit 23-14 **Unimplemented:** Read as ‘1’
- bit 13-6 **FAMID<7:0>:** Device Family Identifier bits
 00010001 = PIC24FJ64GA004 family
- bit 5-0 **DEV<5:0>:** Individual Device Identifier bits
 000100 = PIC24FJ16GA002
 000101 = PIC24FJ32GA002
 000110 = PIC24FJ48GA002
 000111 = PIC24FJ64GA002
 001100 = PIC24FJ16GA004
 001101 = PIC24FJ32GA004
 001110 = PIC24FJ48GA004
 001111 = PIC24FJ64GA004

PIC24FJ64GA004 FAMILY

25.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

25.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

PIC24FJ64GA004 FAMILY

FIGURE 27-5: CLKO AND I/O TIMING CHARACTERISTICS

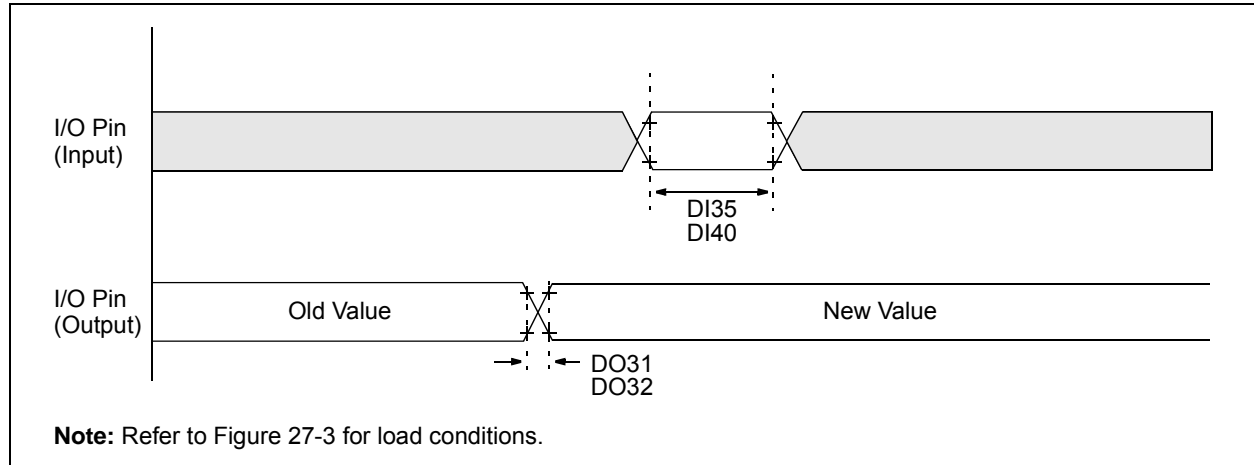


TABLE 27-19: CLKO AND I/O TIMING REQUIREMENTS

AC 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				
Param No.	Sym	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
DO31	TioR	Port Output Rise Time	—	10	25	ns	
DO32	TioF	Port Output Fall Time	—	10	25	ns	
DI35	TINP	INTx Pin High or Low Time (output)	20	—	—	ns	
DI40	TRBP	CNx High or Low Time (input)	2	—	—	Tcy	

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

PIC24FJ64GA004 FAMILY

TABLE 27-21: A/D CONVERSION TIMING REQUIREMENTS⁽¹⁾

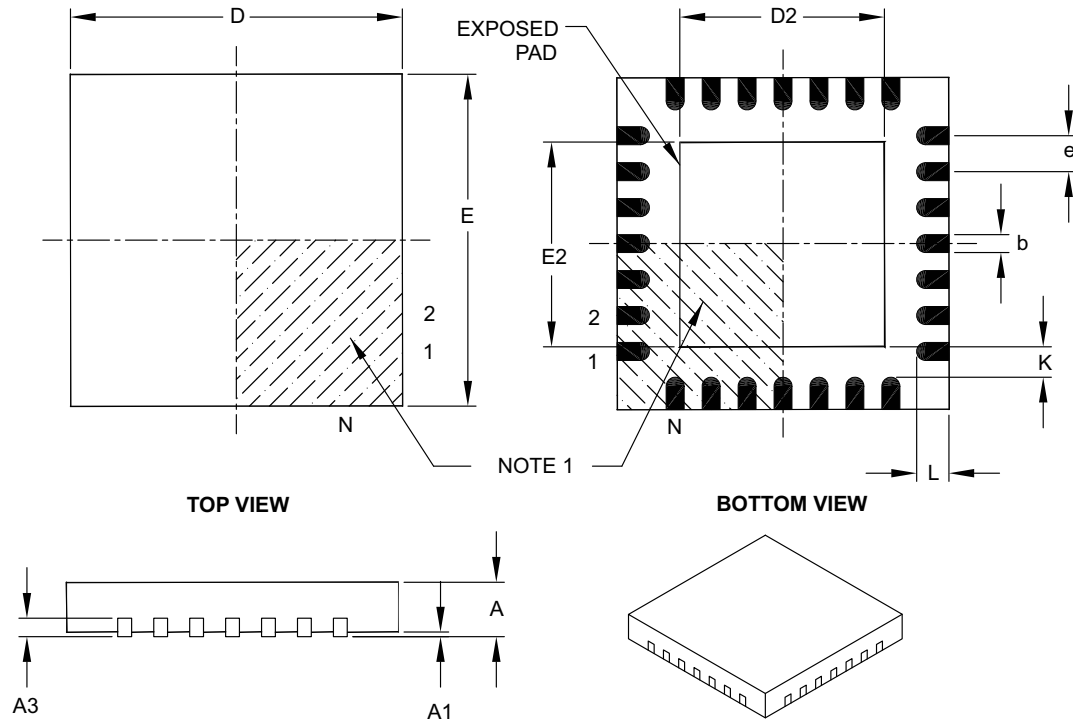
AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)				
			Operating temperature				
			-40°C ≤ TA ≤ +85°C for Industrial				
			-40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ	Max.	Units	Conditions
Clock Parameters							
AD50	TAD	A/D Clock Period	75	—	—	ns	Tcy = 75 ns, AD1CON3 in default state
AD51	tRC	A/D Internal RC Oscillator Period	—	250	—	ns	
Conversion Rate							
AD55	tCONV	Conversion Time	—	12	—	TAD	
AD56	FCNV	Throughput Rate	—	—	500	ksps	AVDD ≥ 2.7V
AD57	tsAMP	Sample Time	—	1	—	TAD	
Clock Parameters							
AD61	tpSS	Sample Start Delay from Setting Sample bit (SAMP)	2	—	3	TAD	

Note 1: Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.

PIC24FJ64GA004 FAMILY

28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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 Pins	N	28		
Pitch	e	0.65 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	6.00 BSC		
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	3.65	3.70	4.20
Contact Width	b	0.23	0.30	0.35
Contact Length	L	0.50	0.55	0.70
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-105B