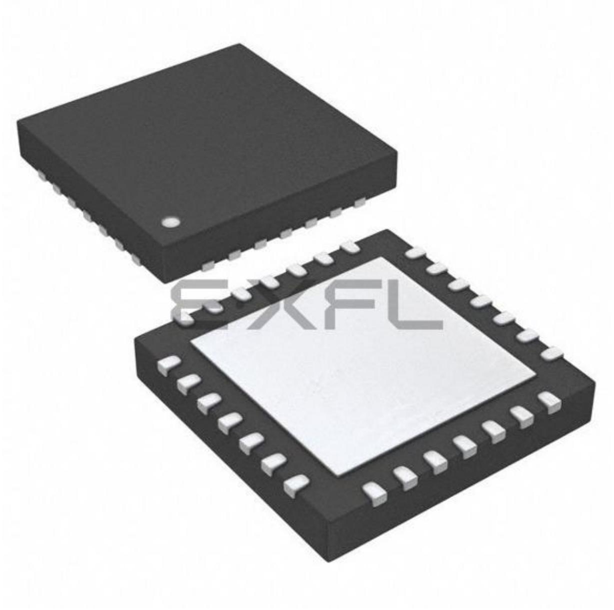


Welcome to E-XFL.COM

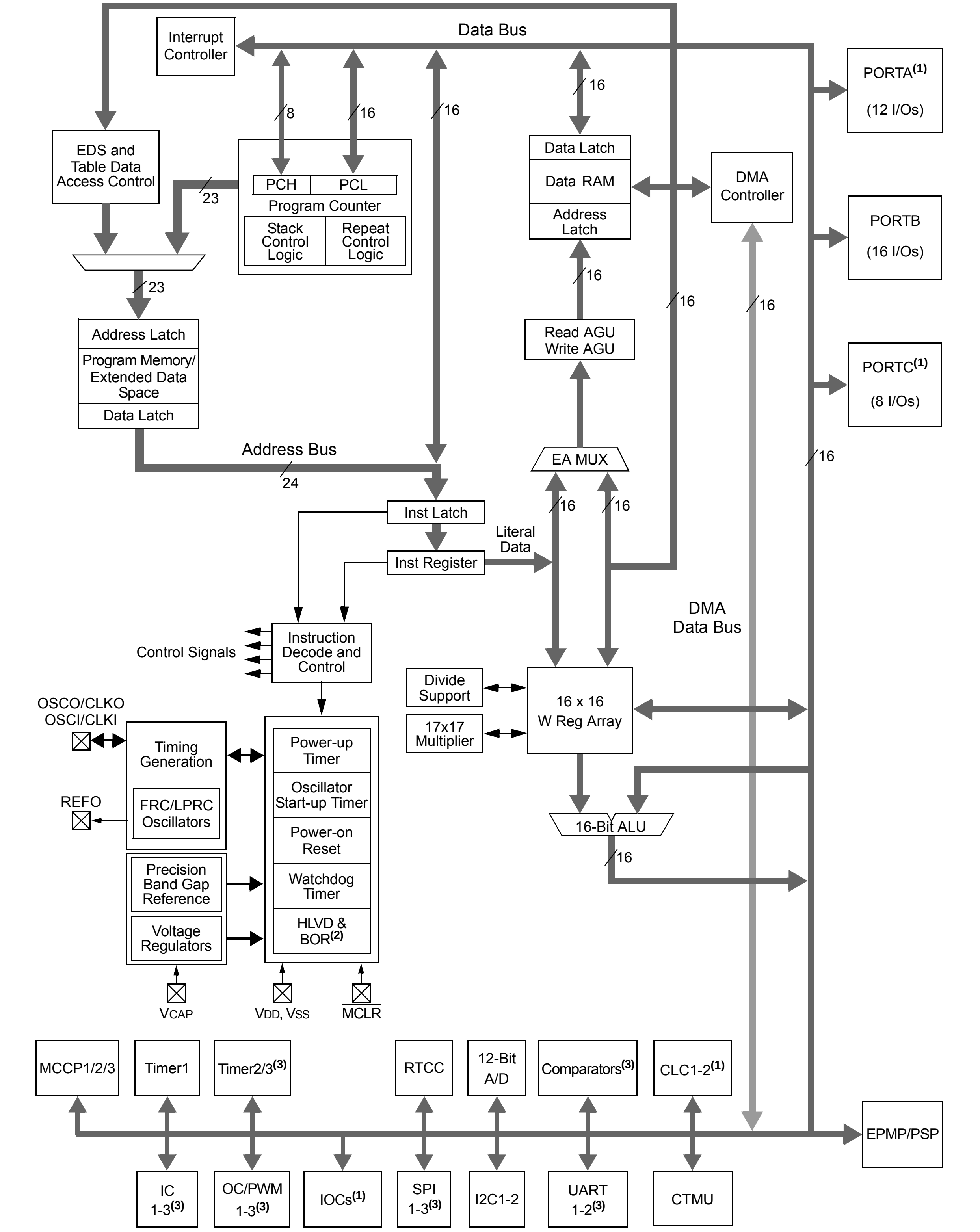
What is "Embedded - Microcontrollers"?

"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	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, I²S, LVD, POR, PWM, WDT
Number of I/O	22
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 10x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj64ga702-i-ml

FIGURE 1-1: PIC24FJ256GA705 FAMILY GENERAL BLOCK DIAGRAM



Note 1: Not all I/O pins or features are implemented on all device pinout configurations. See Table 1-3 for specific implementations by pin count.

2: BOR functionality is provided when the on-board voltage regulator is enabled.

3: Some peripheral I/Os are only accessible through remappable pins.

4.1.4 CODE-PROTECT CONFIGURATION BITS

The device implements intermediate security features defined by the FSEC register. The Boot Segment (BS) is the higher privileged segment and the General Segment (GS) is the lower privileged segment. The total user code memory can be split into BS or GS. The size of the segments is determined by the BSLIM<12:0> bits. The relative location of the segments within user space does not change, such that BS (if present) occupies the memory area just after the Interrupt Vector Table (IVT) and the GS occupies the space just after the BS (or if the Alternate IVT is enabled, just after it).

The Configuration Segment (CS) is a small segment (less than a page, typically just one row) within user Flash address space. It contains all user configuration data that is loaded by the NVM Controller during the Reset sequence.

4.1.5 CUSTOMER OTP MEMORY

PIC24FJ256GA705 family devices provide 256 bytes of One-Time-Programmable (OTP) memory, located at addresses, 801700h through 8017FEh. This memory can be used for persistent storage of application-specific information that will not be erased by reprogramming the device. This includes many types of information, such as (but not limited to):

- Application Checksums
- Code Revision Information
- Product Information
- Serial Numbers
- System Manufacturing Dates
- Manufacturing Lot Numbers

Customer OTP memory may be programmed in any mode, including user RTSP mode, but it cannot be erased. Data is not cleared by a chip erase.

Note:	Do not write the OTP memory more than one time. Writing to the OTP memory more than once may result in a permanent ECC Double-Bit Error (ECCDBE) trap.

TABLE 4-13: EDS MEMORY ADDRESS WITH DIFFERENT PAGES AND ADDRESSES

DSRPAG (Data Space Read Register)	DSWPAG (Data Space Write Register)	Source/Destination Address while Indirect Addressing	24-Bit EA Pointing to EDS	Comment
x[1]	x[1]	0000h to 1FFFh	000000h to 001FFFh	Near Data Space[2]
		2000h to 7FFFh	002000h to 007FFFh	
001h	001h	8000h to FFFFh	008000h to 00FFFEh	EPMP Memory Space
002h	002h		010000h to 017FFEh	
003h • • • • • 1FFh	003h • • • • • 1FFh		018000h to 0187FEh • • • • FF8000h to FFFFFEh	
000h	000h		Invalid Address	Address Error Trap[3]

Note 1: If the source/destination address is below 8000h, the DSRPAG and DSWPAG registers are not considered.

2: This Data Space can also be accessed by Direct Addressing.

3: When the source/destination address is above 8000h and DSRPAG/DSWPAG are '0', an address error trap will occur.

4.2.6 SOFTWARE STACK

Apart from its use as a Working register, the W15 register in PIC24F devices is also used as a Software Stack Pointer (SSP). The pointer always points to the first available free word and grows from lower to higher addresses. It pre-decrements for stack pops and post-increments for stack pushes, as shown in Figure 4-6. Note that for a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

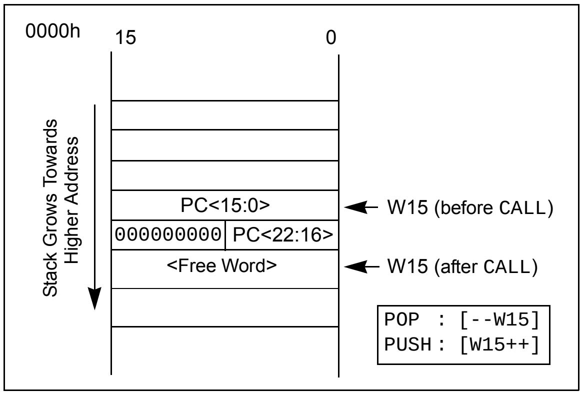
Note:	A PC push during exception processing will concatenate the SRL register to the MSB of the PC prior to the push.

The Stack Pointer Limit Value register (SPLIM), associated with the Stack Pointer, sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' as all stack operations must be word-aligned. Whenever an EA is generated using W15 as a source or destination pointer, the resulting address is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal, and a push operation is performed, a stack error trap will not occur. The stack error trap will occur on a subsequent push operation. Thus, for example, if it is desirable to cause a stack error trap when the stack grows beyond address 2000h in RAM, initialize the SPLIM with the value, 1FFEh.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0800h. This prevents the stack from interfering with the SFR space.

A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

FIGURE 4-6: CALL STACK FRAME



REGISTER 5-3: DMAINTn: DMA CHANNEL n INTERRUPT REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DBUFWF[1]	CHSEL6	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
HIGHIF[1,2]	LOWIF[1,2]	DONEIF[1]	HALFIF[1]	OVRUNIF[1]	—	—	HALFEN
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 **DBUFWF:** DMA Buffered Data Write Flag bit[1]

1 = The content of the DMA buffer has not been written to the location specified in DMADSTn or DMASRCn in Null Write mode
0 = The content of the DMA buffer has been written to the location specified in DMADSTn or DMASRCn in Null Write mode

bit 14-8 **CHSEL<6:0>:** DMA Channel Trigger Selection bits

See Table 5-1 for a complete list.

bit 7 **HIGHIF:** DMA High Address Limit Interrupt Flag bit[1,2]

1 = The DMA channel has attempted to access an address higher than DMAH or the upper limit of the data RAM space
0 = The DMA channel has not invoked the high address limit interrupt

bit 6 **LOWIF:** DMA Low Address Limit Interrupt Flag bit[1,2]

1 = The DMA channel has attempted to access the DMA SFR address lower than DMAL, but above the SFR range (07FFh)
0 = The DMA channel has not invoked the low address limit interrupt

bit 5 **DONEIF:** DMA Complete Operation Interrupt Flag bit[1]

If CHEN = 1:
1 = The previous DMA session has ended with completion
0 = The current DMA session has not yet completed

If CHEN = 0:
1 = The previous DMA session has ended with completion
0 = The previous DMA session has ended without completion

bit 4 **HALFIF:** DMA 50% Watermark Level Interrupt Flag bit[1]

1 = DMACNTn has reached the halfway point to 0000h
0 = DMACNTn has not reached the halfway point

bit 3 **OVRUNIF:** DMA Channel Overrun Flag bit[1]

1 = The DMA channel is triggered while it is still completing the operation based on the previous trigger
0 = The overrun condition has not occurred

bit 2-1 **Unimplemented:** Read as ‘0’

bit 0 **HALFEN:** Halfway Completion Watermark bit

1 = Interrupts are invoked when DMACNTn has reached its halfway point and at completion
0 = An interrupt is invoked only at the completion of the transfer

Note 1: Setting these flags in software does not generate an interrupt.

2: Testing for address limit violations (DMASRCn or DMADSTn is either greater than DMAH or less than DMAL) is NOT done before the actual access.

TABLE 8-2: INTERRUPT VECTOR DETAILS

Interrupt Source	IRQ #	IVT Address	Interrupt Bit Location		
			Flag	Enable	Priority
Highest Natural Order Priority					
INT0 – External Interrupt 0	0	000014h	IFS0<0>	IEC0<0>	INT0Interrupt
IC1 – Input Capture 1	1	000016h	IFS0<1>	IEC0<1>	IC1Interrupt
OC1 – Output Compare 1	2	000018h	IFS0<2>	IEC0<2>	OC1Interrupt
T1 – Timer1	3	00001Ah	IFS0<3>	IEC0<3>	T1Interrupt
DMA0 – Direct Memory Access 0	4	00001Ch	IFS0<4>	IEC0<4>	DMA0Interrupt
IC2 – Input Capture 2	5	00001Eh	IFS0<5>	IEC0<5>	IC2Interrupt
OC2 – Output Compare 2	6	000020h	IFS0<6>	IEC0<6>	OC2Interrupt
T2 – Timer2	7	000022h	IFS0<7>	IEC0<7>	T2Interrupt
T3 – Timer3	8	000024h	IFS0<8>	IEC0<8>	T3Interrupt
SPI1 – SPI1 General	9	000026h	IFS0<9>	IEC0<9>	SPI1Interrupt
SPI1TX – SPI1 Transfer Done	10	000028h	IFS0<10>	IEC0<10>	SPI1TXInterrupt
U1RX – UART1 Receiver	11	00002Ah	IFS0<11>	IEC0<11>	U1RXInterrupt
U1TX – UART1 Transmitter	12	00002Ch	IFS0<12>	IEC0<12>	U1TXInterrupt
ADC1 – A/D Converter 1	13	00002Eh	IFS0<13>	IEC0<13>	ADC1Interrupt
DMA1 – Direct Memory Access 1	14	000030h	IFS0<14>	IEC0<14>	DMA1Interrupt
NVM – NVM Program/Erase Complete	15	000032h	IFS0<15>	IEC0<15>	NVMInterrupt
SI2C1 – I2C1 Slave Events	16	000034h	IFS1<0>	IEC1<0>	SI2C1Interrupt
MI2C1 – I2C1 Master Events	17	000036h	IFS1<1>	IEC1<1>	MI2C1Interrupt
Comp – Comparator	18	000038h	IFS1<2>	IEC1<2>	CompInterrupt
IOC – Interrupt-on-Change Interrupt	19	00003Ah	IFS1<3>	IEC1<3>	IOCInterrupt
INT1 – External Interrupt 1	20	00003Ch	IFS1<4>	IEC1<4>	INT1Interrupt
—	21	—	—	—	—
—	22	—	—	—	—
—	23	—	—	—	—
DMA2 – Direct Memory Access 2	24	000044h	IFS1<8>	IEC1<8>	DMA2Interrupt
OC3 – Output Compare 3	25	000046h	IFS1<9>	IEC1<9>	OC3Interrupt
—	26	—	—	—	—
—	27	—	—	—	—
—	28	—	—	—	—
INT2 – External Interrupt 2	29	00004Eh	IFS1<13>	IEC1<13>	INT2Interrupt
U2RX – UART2 Receiver	30	000050h	IFS1<14>	IEC1<14>	U2RXInterrupt
U2TX – UART2 Transmitter	31	000052h	IFS1<15>	IEC1<15>	U2TXInterrupt
SPI2 – SPI2 General	32	000054h	IFS2<0>	IEC2<0>	SPI2Interrupt
SPI2TX – SPI2 Transfer Done	33	000056h	IFS2<1>	IEC2<1>	SPI2TXInterrupt
—	34	—	—	—	—
—	35	—	—	—	—
DMA3 – Direct Memory Access 3	36	00005Ch	IFS2<4>	IEC2<4>	DMA3Interrupt
IC3 – Input Capture 3	37	00005Eh	IFS2<5>	IEC2<5>	IC3Interrupt
—	38	—	—	—	—
—	39	—	—	—	—
—	40	—	—	—	—
CCT3 – Capture/Compare Timer3	43	00006Ah	IFS2<11>	IEC2<11>	CCT3Interrupt

9.5 Oscillator Modes

The PLL block is shown in Figure 9-2. In this system, the input from the Primary Oscillator is divided down by a PLL prescaler to generate a 4 MHz output. This is used to drive an on-chip, 96 MHz PLL frequency multiplier to drive the fixed, divide-by-3 frequency divider and configurable PLL prescaler/divider to generate a range of system clock frequencies. The CPDIV<1:0> bits select the system clock speed. Available clock options are listed in Table 9-2.

The user must manually configure the PLL divider to generate the required 4 MHz output using the PLLMODE<3:0> Configuration bits. This limits the choices for Primary Oscillator frequency to a total of eight possibilities, as shown in Table 9-3.

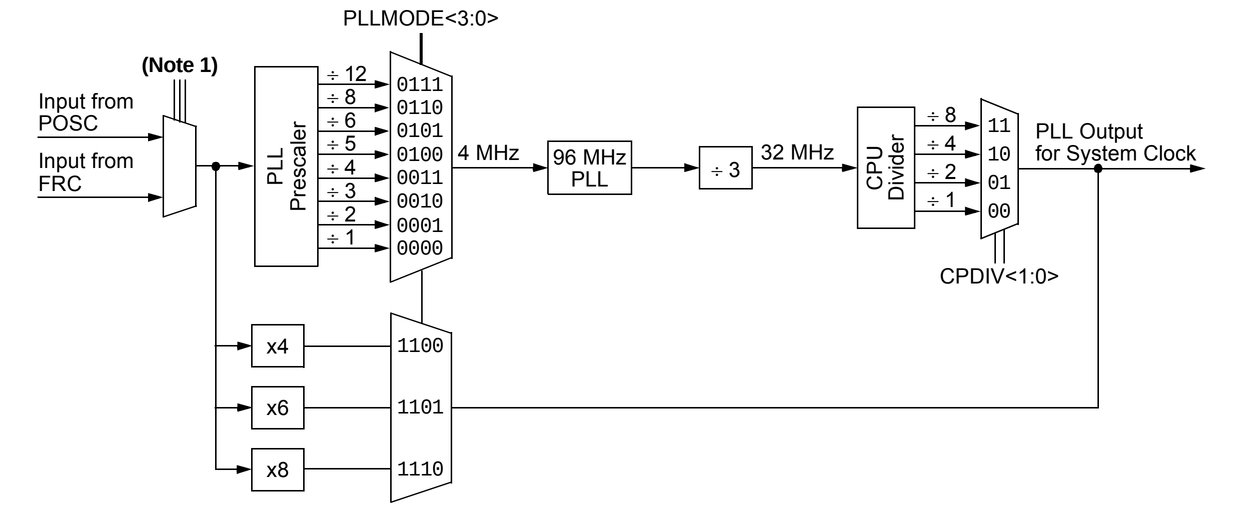
TABLE 9-2: SYSTEM CLOCK OPTIONS

MCU Clock Division (CPDIV<1:0>)	Microcontroller Clock Frequency
None (00)	32 MHz
÷2 (01)	16 MHz
÷4 (10)	8 MHz
÷8 (11)	4 MHz

TABLE 9-3: VALID PRIMARY OSCILLATOR CONFIGURATIONS

Input Oscillator Frequency	Clock Mode	PLL Mode (PLLMODE<3:0>)
48 MHz	ECPLL	÷12 (0111)
32 MHz	HSPLL, ECPLL	÷8 (0110)
24 MHz	HSPLL, ECPLL	÷6 (0101)
20 MHz	HSPLL, ECPLL	÷5 (0100)
16 MHz	HSPLL, ECPLL	÷4 (0011)
12 MHz	HSPLL, ECPLL	÷3 (0010)
8 MHz	ECPLL, XTPLL, FRCPLL	÷2 (0001)
4 MHz	ECPLL, XTPLL, FRCPLL	÷1 (0000)

FIGURE 9-2: PLL BLOCK



Note 1: This MUX is controlled by the COSC<2:0> bits when running from the PLL or the NOSC<2:0> bits when preparing to switch to the PLL.

REGISTER 11-17: RPINR5: PERIPHERAL PIN SELECT INPUT REGISTER 5

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	ICM2R5	ICM2R4	ICM2R3	ICM2R2	ICM2R1	ICM2R0
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
—	—	ICM1R5	ICM1R4	ICM1R3	ICM1R2	ICM1R1	ICM1R0
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 **ICM2R<5:0>:** Input Capture Mode 2 bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **ICM1R<5:0>:** Input Capture Mode 1 bits

REGISTER 11-18: RPINR6: PERIPHERAL PIN SELECT INPUT REGISTER 6

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	ICM4R5	ICM4R4	ICM4R3	ICM4R2	ICM4R1	ICM4R0
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
—	—	ICM3R5	ICM3R4	ICM3R3	ICM3R2	ICM3R1	ICM3R0
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 **ICM4R<5:0>:** Input Capture Mode 4 bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **ICM3R<5:0>:** Input Capture Mode 3 bits

REGISTER 11-38: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP13R5	RP13R4	RP13R3	RP13R2	RP13R1	RP13R0
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
—	—	RP12R5	RP12R4	RP12R3	RP12R2	RP12R1	RP12R0
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 **RP13R<5:0>:** RP13 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP13 (see Table 11-7 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **RP12R<5:0>:** RP12 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP12 (see Table 11-7 for peripheral function numbers).

REGISTER 11-39: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP15R5	RP15R4	RP15R3	RP15R2	RP15R1	RP15R0
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
—	—	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0
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 **RP15R<5:0>:** RP15 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP15 (see Table 11-7 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **RP14R<5:0>:** RP14 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP14 (see Table 11-7 for peripheral function numbers).

13.0 TIMER2/3

> **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 *"dsPIC33/PIC24 Family Reference Manual"*, **"Timers"** (DS39704), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The Timer2/3 module is a 32-bit timer, which can also be configured as independent, 16-bit timers with selectable operating modes.

As a 32-bit timer, Timer2/3 can operate in three modes:

- Two Independent 16-Bit Timers with All 16-Bit Operating modes (except Asynchronous Counter mode)
- Single 32-Bit Timer
- Single 32-Bit Synchronous Counter

They also support these features:

- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation during Idle and Sleep modes
- Interrupt on a 32-Bit Period Register Match
- A/D Event Trigger (on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode)

Individually, all of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the A/D event trigger. This trigger is implemented only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON and T3CON registers. T2CON is shown in generic form in Register 13-1; T3CON is shown in Register 13-2.

For 32-bit timer/counter operation, Timer2 is the least significant word; Timer3 is the most significant word of the 32-bit timer.

> **Note:** For 32-bit operation, T3CON control bits are ignored. Only T2CON control bits are used for setup and control. Timer2 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3 interrupt flags.

To configure Timer2/3 for 32-bit operation:

1. Set the T32 bit (T2CON<3> = 1).
2. Select the prescaler ratio for Timer2 using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. If TCS is set to an external clock, RPINRx (TyCK) must be configured to an available RPn/RPIn pin. For more information, see **Section 11.5 "Peripheral Pin Select (PPS)"**.
4. Load the timer period value. PR3 will contain the most significant word (msw) of the value, while PR2 contains the least significant word (lsw).
5. If interrupts are required, set the interrupt enable bit, T3IE. Use the priority bits, T3IP<2:0>, to set the interrupt priority. Note that while Timer2 controls the timer, the interrupt appears as a Timer3 interrupt.
6. Set the TON bit (= 1).

The timer value, at any point, is stored in the register pair, TMR<3:2>. TMR3 always contains the most significant word of the count, while TMR2 contains the least significant word.

To configure any of the timers for individual 16-bit operation:

1. Clear the T32 bit (T2CON<3>).
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. See **Section 11.5 "Peripheral Pin Select (PPS)"** for more information.
4. Load the timer period value into the PRx register.
5. If interrupts are required, set the interrupt enable bit, TxIE. Use the priority bits, TxIP<2:0>, to set the interrupt priority.
6. Set the TON (TxCON<15> = 1) bit.

REGISTER 16-7: CCPxSTATL: CCPx STATUS REGISTER LOW

U-0	U-0	U-0	U-0	U-0	W-0	U-0	U-0
—	—	—	—	—	ICGARM	—	—
bit 15							bit 8

R-0	W1-0	W1-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
CCPTRIG	TRSET	TRCLR	ASEVT	SCEVT	ICDIS	ICOV	ICBNE
bit 7							bit 0

Legend:	C = Clearable bit	W = Writable bit	
R = Readable bit	W1 = Write '1' Only 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 **ICGARM:** Input Capture Gate Arm bit

A write of '1' to this location will arm the Input Capture x module for a one-shot gating event when ICGSM<1:0> = 01 or 10; read as '0'.

bit 9-8 **Unimplemented:** Read as '0'

bit 7 **CCPTRIG:** CCPx Trigger Status bit

1 = Timer has been triggered and is running
0 = Timer has not been triggered and is held in Reset

bit 6 **TRSET:** CCPx Trigger Set Request bit

Writes '1' to this location to trigger the timer when TRIGEN = 1 (location always reads as '0').

bit 5 **TRCLR:** CCPx Trigger Clear Request bit

Writes '1' to this location to cancel the timer trigger when TRIGEN = 1 (location always reads as '0').

bit 4 **ASEVT:** CCPx Auto-Shutdown Event Status/Control bit

1 = A shutdown event is in progress; CCPx outputs are in the shutdown state
0 = CCPx outputs operate normally

bit 3 **SCEVT:** Single Edge Compare Event Status bit

1 = A single edge compare event has occurred
0 = A single edge compare event has not occurred

bit 2 **ICDIS:** Input Capture x Disable bit

1 = Event on Input Capture x pin (ICMx) does not generate a capture event
0 = Event on Input Capture x pin will generate a capture event

bit 1 **ICOV:** Input Capture x Buffer Overflow Status bit

1 = The Input Capture x FIFO buffer has overflowed
0 = The Input Capture x FIFO buffer has not overflowed

bit 0 **ICBNE:** Input Capture x Buffer Status bit

1 = Input Capture x buffer has data available
0 = Input Capture x buffer is empty

18.2 Setting Baud Rate When Operating as a Bus Master

To compute the Baud Rate Generator reload value, use Equation 18-1.

EQUATION 18-1: COMPUTING BAUD RATE RELOAD VALUE[(1,2,3)]

$$F_{SCL} = \frac{F_{CY}}{(I2CxBRG + 2) * 2}$$

or:

$$I2CxBRG = \left[\frac{F_{CY}}{(F_{SCL} * 2)} - 2 \right]$$

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

2: These clock rate values are for guidance only. The actual clock rate can be affected by various system-level parameters. The actual clock rate should be measured in its intended application.

3: BRG values of 0 and 1 are forbidden.

18.3 Slave Address Masking

The I2CxMSK register (Register 18-4) designates address bit positions as "don't care" for both 7-Bit and 10-Bit Addressing modes. Setting a particular bit location (= `1`) in the I2CxMSK register causes the slave module to respond, whether the corresponding address bit value is a '`0`' or a '`1`'. For example, when I2CxMSK is set to '`0010000000`', the slave module will detect both addresses, '`0000000000`' and '`0010000000`'.

To enable address masking, the Intelligent Peripheral Management Interface (IPMI) must be disabled by clearing the STRICT bit (I2CxCONL<11>).

Note: As a result of changes in the I^{2}C protocol, the addresses in Table 18-2 are reserved and will not be Acknowledged in Slave mode. This includes any address mask settings that include any of these addresses.

TABLE 18-1: I2Cx CLOCK RATES[(1,2)]

Required System FSCL	FCY	I2CxBRG Value		Actual FSCL
		(Decimal)	(Hexadecimal)	
100 kHz	16 MHz	78	4E	100 kHz
100 kHz	8 MHz	38	26	100 kHz
100 kHz	4 MHz	18	12	100 kHz
400 kHz	16 MHz	18	12	400 kHz
400 kHz	8 MHz	8	8	400 kHz
400 kHz	4 MHz	3	3	400 kHz
1 MHz	16 MHz	6	6	1.000 MHz
1 MHz	8 MHz	2	2	1.000 MHz

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

2: These clock rate values are for guidance only. The actual clock rate can be affected by various system-level parameters. The actual clock rate should be measured in its intended application.

TABLE 18-2: I2Cx RESERVED ADDRESSES[(1)]

Slave Address	R/$\overline{W}$ Bit	Description
`0000 000`	`0`	General Call Address[(2)]
`0000 000`	`1`	Start Byte
`0000 001`	`x`	Cbus Address
`0000 01x`	`x`	Reserved
`0000 1xx`	`x`	HS Mode Master Code
`1111 0xx`	`x`	10-Bit Slave Upper Byte[(3)]
`1111 1xx`	`x`	Reserved

Note 1: The address bits listed here will never cause an address match independent of address mask settings.

2: This address will be Acknowledged only if GCEN = `1`.

3: A match on this address can only occur on the upper byte in 10-Bit Addressing mode.

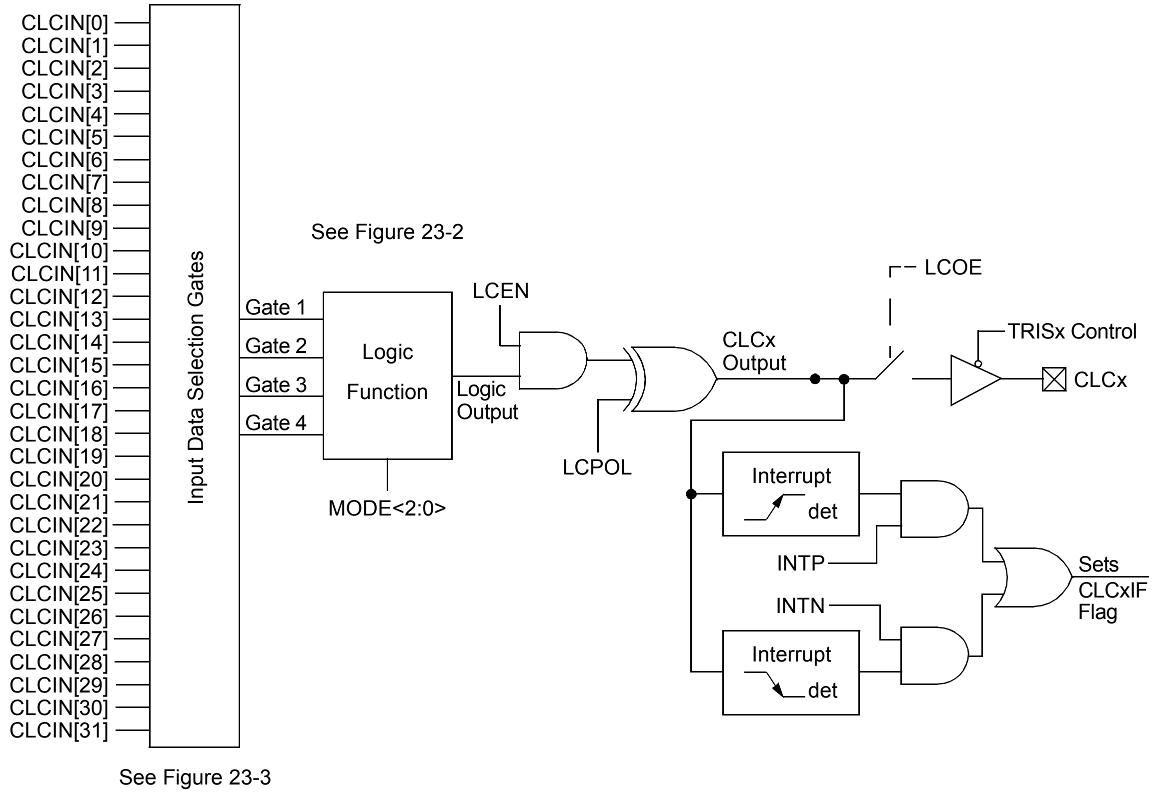
23.0 CONFIGURABLE LOGIC CELL (CLC) GENERATOR

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 *"dsPIC33/PIC24 Family Reference Manual"*, **"Configurable Logic Cell (CLC)"** (DS33949), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The Configurable Logic Cell (CLC) module allows the user to specify combinations of signals as inputs to a logic function and to use the logic output to control other peripherals or I/O pins. This provides greater flexibility and potential in embedded designs, since the CLC module can operate outside the limitations of software execution and supports a vast amount of output designs.

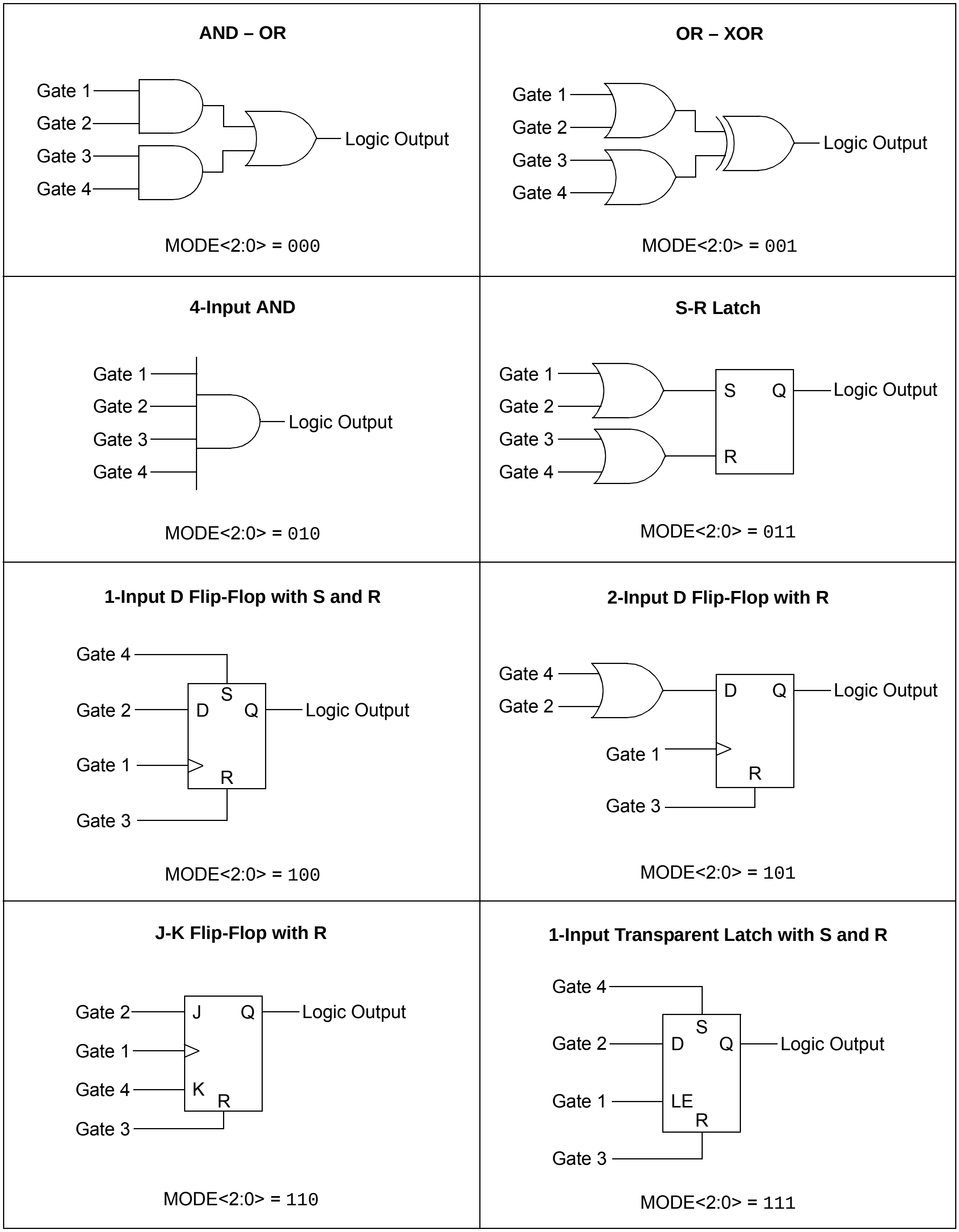
There are four input gates to the selected logic function. These four input gates select from a pool of up to 32 signals that are selected using four data source selection multiplexers. Figure 23-1 shows an overview of the module. Figure 23-3 shows the details of the data source multiplexers and logic input gate connections.

FIGURE 23-1: CLCx MODULE



Note: All register bits shown in this figure can be found in the CLCxCONL register.

FIGURE 23-2: CLCx LOGIC FUNCTION COMBINATORIAL OPTIONS



AND – OR

MODE<2:0> = `000`

OR – XOR

MODE<2:0> = `001`

4-Input AND

MODE<2:0> = `010`

S-R Latch

MODE<2:0> = `011`

1-Input D Flip-Flop with S and R

MODE<2:0> = `100`

2-Input D Flip-Flop with R

MODE<2:0> = `101`

J-K Flip-Flop with R

MODE<2:0> = `110`

1-Input Transparent Latch with S and R

MODE<2:0> = `111`

REGISTER 24-9: AD1CSSH: A/D INPUT SCAN SELECT REGISTER (HIGH WORD)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	CSS<28:24>				
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **CSS<28:24>:** A/D Input Scan Selection bits
1 = Includes corresponding channel for input scan
0 = Skips channel for input scan

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 24-10: AD1CSSL: A/D INPUT SCAN SELECT REGISTER (LOW WORD)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS<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 **CSS<15:0>:** A/D Input Scan Selection bits
1 = Includes corresponding channel for input scan
0 = Skips channel for input scan

REGISTER 29-6: FWDT CONFIGURATION REGISTER

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 23							bit 16

U-1	R/PO-1	R/PO-1	U-1	R/PO-1	U-1	R/PO-1	R/PO-1
—	WDTCLK1	WDTCLK0	—	WDTCMX	—	WDTWIN1	WDTWIN0
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
WINDIS	FWDTEN1	FWDTEN0	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

Legend:	PO = Program Once 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 23-15 **Unimplemented:** Read as ‘1’

bit 14-13 **WDTCLK<1:0>:** Watchdog Timer Clock Select bits (when WDTCMX = 1)

11 = Always uses LPRC
10 = Uses FRC when WINDIS = 0, system clock is not LPRC and device is not in Sleep; otherwise, uses LPRC
01 = Always uses SOSC
00 = Uses peripheral clock when system clock is not LPRC and device is not in Sleep; otherwise, uses LPRC

bit 12 **Unimplemented:** Read as ‘1’

bit 11 **WDTCMX:** WDT Clock MUX Control bit

1 = Enables WDT clock MUX, WDT clock is selected by WDTCLK<1:0>
0 = WDT clock is LPRC

bit 10 **Unimplemented:** Read as ‘1’

bit 9-8 **WDTWIN<1:0>:** Watchdog Timer Window Width bits

11 = WDT window is 25% of the WDT period
10 = WDT window is 37.5% of the WDT period
01 = WDT window is 50% of the WDT period
00 = WDT window is 75% of the WDT period

bit 7 **WINDIS:** Windowed Watchdog Timer Disable bit

1 = Windowed WDT is disabled
0 = Windowed WDT is enabled

bit 6-5 **FWDTEN<1:0>:** Watchdog Timer Enable bits

11 = WDT is enabled
10 = WDT is disabled (control is placed on the SWDTEN bit)
01 = WDT is enabled only while device is active and disabled in Sleep; SWDTEN bit is disabled
00 = WDT and SWDTEN are disabled

bit 4 **FWPSA:** Watchdog Timer Prescaler bit

1 = WDT prescaler ratio of 1:128
0 = WDT prescaler ratio of 1:32

REGISTER 29-6: FWDT CONFIGURATION REGISTER (CONTINUED)

bit 3-0 **WDTPS<3:0>:** Watchdog Timer Postscale Select bits

`1111` = 1:32,768
`1110` = 1:16,384
`1101` = 1:8,192
`1100` = 1:4,096
`1011` = 1:2,048
`1010` = 1:1,024
`1001` = 1:512
`1000` = 1:256
`0111` = 1:128
`0110` = 1:64
`0101` = 1:32
`0100` = 1:16
`0011` = 1:8
`0010` = 1:4
`0001` = 1:2
`0000` = 1:1

30.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

30.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

30.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

30.9 PICkit 3 In-Circuit Debugger/Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

30.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

TABLE 32-25: A/D CONVERSION TIMING REQUIREMENTS[1]

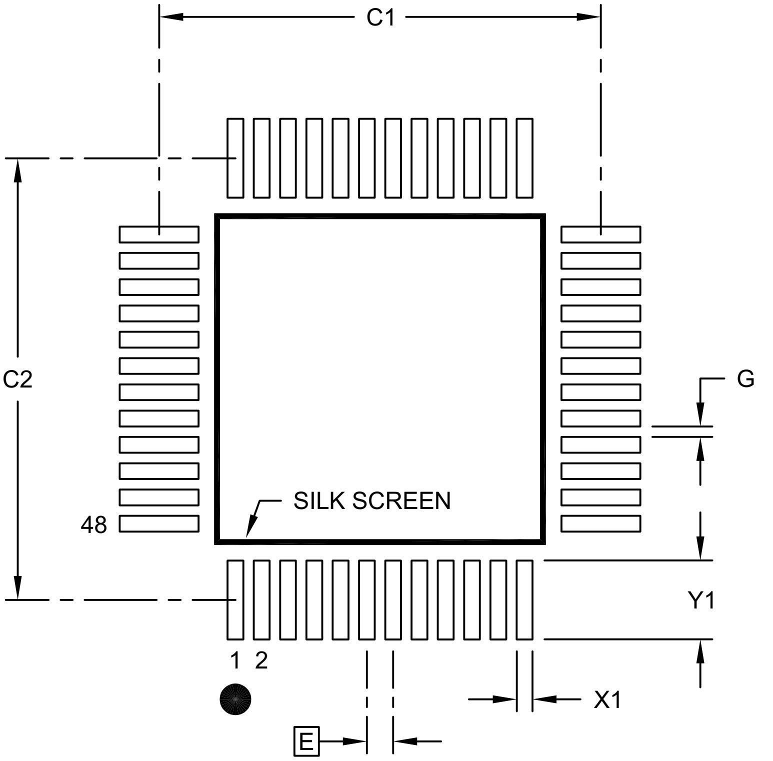
AC CHARACTERISTICS			**Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
Clock Parameters							
AD50	TAD	A/D Clock Period	278	—	—	ns	
AD51	tRC	A/D Internal RC Oscillator Period	—	250	—	ns	
Conversion Rate							
AD55	tCONV	SAR Conversion Time, 12-Bit Mode	—	14	—	TAD	
AD55A		SAR Conversion Time, 10-Bit Mode is Typical 12 TAD	—	12	—	TAD	
AD56	FCNV	Throughput Rate	—	—	200	ksps	AVDD > 2.7V[2]
AD57	tSAMP	Sample Time	—	1	—	TAD	
Clock Synchronization							
AD61	tPSS	Sample Start Delay from Setting Sample bit (SAMP)	1.5	—	2.5	TAD	

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

2: Throughput rate is based on AD55 + AD57 + AD61 and the period of TAD.

48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		8.40	
Contact Pad Spacing	C2		8.40	
Contact Pad Width (X48)	X1			0.30
Contact Pad Length (X48)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2300-PT Rev A

NOTES: