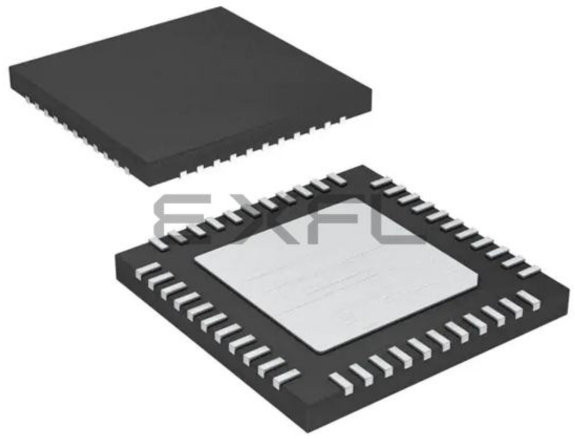


Welcome to E-XFL.COM

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, PMP/PSP, SmartCard, SPI, UART/USART
Peripherals	AES, Brown-out Detect/Reset, DMA, I²S, HLVD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 13x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj64ga204-e-ml

TABLE 1-3: PIC24FJ128GA204 FAMILY PINOUT DESCRIPTIONS

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	28-Pin SPDIP/SOIC/SSOP	28-Pin QFN-S	44-Pin TQFP/QFN			
AN0	2	27	19	I	ANA	12-Bit SAR A/D Converter Inputs.
AN1	3	28	20	I	ANA	
AN2	4	1	21	I	ANA	
AN3	5	2	22	I	ANA	
AN4	6	3	23	I	ANA	
AN5	7	4	24	I	ANA	
AN6	25	22	14	I	ANA	
AN7	24	21	11	I	ANA	
AN8	23	20	10	I	ANA	
AN9	26	23	15	I	ANA	
AN10	—	—	25	I	ANA	
AN11	—	—	26	I	ANA	
AN12	—	—	27	I	ANA	
ASCL1	15	12	42	—	—	
ASDA1	2	27	19	—	—	
AVDD	—	—	17	P	ANA	Positive Supply for Analog modules.
AVSS	—	24	16	P	ANA	Ground Reference for Analog modules.
C1INA	7	4	24	I	ANA	Comparator 1 Input A.
C1INB	6	3	23	I	ANA	Comparator 1 Input B.
C1INC	24	15	1	I	ANA	Comparator 1 Input C.
C1IND	9	6	30	I	ANA	Comparator 1 Input D.
C2INA	5	2	22	I	ANA	Comparator 2 Input A.
C2INB	4	1	21	I	ANA	Comparator 2 Input B.
C2INC	18	15	1	I	ANA	Comparator 2 Input C.
C2IND	10	7	31	I	ANA	Comparator 2 Input D.
C3INA	26	23	15	I	ANA	Comparator 3 Input A.
C3INB	25	22	14	I	ANA	Comparator 3 Input B.
C3INC	2	15	1	I	ANA	Comparator 3 Input C.
C3IND	3	28	20	I	ANA	Comparator 3 Input D.
CLKI	9	6	30	I	ANA	Main Clock Input Connection.
CLKO	10	7	31	O	—	System Clock Output.

Legend: ST = Schmitt Trigger input; TTL = TTL compatible input; I = Input; ANA = Analog input; O = Output; P = Power; I²C = ST with I²C™ or SMBus levels

TABLE 4-3: CPU CORE REGISTERS 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
WREG0	0000	Working Register 0																0000
WREG1	0002	Working Register 1																0000
WREG2	0004	Working Register 2																0000
WREG3	0006	Working Register 3																0000
WREG4	0008	Working Register 4																0000
WREG5	000A	Working Register 5																0000
WREG6	000C	Working Register 6																0000
WREG7	000E	Working Register 7																0000
WREG8	0010	Working Register 8																0000
WREG9	0012	Working Register 9																0000
WREG10	0014	Working Register 10																0000
WREG11	0016	Working Register 11																0000
WREG12	0018	Working Register 12																0000
WREG13	001A	Working Register 13																0000
WREG14	001C	Working Register 14																0000
WREG15	001E	Working Register 15																0800
SPLIM	0020	Stack Pointer Limit Value Register																xxxx
PCL	002E	Program Counter Low Word Register																0000
PCH	0030	—	—	—	—	—	—	—	—	Program Counter High Word Register								0000
DSRPAG	0032	—	—	—	—	—	—	Extended Data Space Read Page Address Register										0001
DSWPAG	0034	—	—	—	—	—	—	—	Extended Data Space Write Page Address Register									0001
RCOUNT	0036	REPEAT Loop Counter Register																xxxx
SR	0042	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	r	—	—	0004
DISICNT	0052	—	—	Disable Interrupts Counter Register														xxxx
TBLPAG	0054	—	—	—	—	—	—	—	—	Table Memory Page Address Register								0000

Legend: — = unimplemented, read as '0'; r = reserved, do not modify; x = unknown value on Reset. Reset values are shown in hexadecimal.

TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP (CONTINUED)

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
IPC16	00C4	—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0	—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—	4440
IPC18	00C8	—	—	—	—	—	—	—	—	—	—	—	—	—	HLVDIP<2:0>			0004
IPC19	00CA	—	—	—	—	—	—	—	—	—	CTMUIP<2:0>			—	—	—	—	0040
IPC20	00CC	—	U3TXIP2	U3TXIP1	U3TXIP0	—	U3RXIP2	U3RXIP1	U3RXIP0	—	U3ERIP2	U3ERIP1	U3ERIP0	—	—	—	—	4440
IPC21	00CE	—	U4ERIP2	U4ERIP1	U4ERIP0	—	—	—	—	—	I2C2BCIP2	I2C2BCIP1	I2C2BCIP0	—	I2C1BCIP2	I2C1BCIP1	I2C1BCIP0	4044
IPC22	00D0	—	SPI3TXIP2	SPI3TXIP1	SPI3TXIP0	—	SPI3IP2	SPI3IP1	SPI3IP0	—	U4TXIP2	U4TXIP1	U4TXIP0	—	U4RXIP2	U4RXIP1	U4RXIP0	4444
IPC26	00D8	—	—	—	—	—	FSTIP<2:0>			—	—	—	—	—	—	—	—	0400
IPC29	00DE	—	—	—	—	—	—	—	—	—	JTAGIP<2:0>			—	—	—	—	0040
INTTREG	00E0	CPUIRQ	r	VHOLD	—	ILR3	ILR2	ILR1	ILR0	VECNUM7	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0	0000

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

REGISTER 5-1: DMACON: DMA ENGINE CONTROL REGISTER

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
DMAEN	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PRSSEL
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 **DMAEN:** DMA Module Enable bit
1 = Enables module
0 = Disables module and terminates all active DMA operation(s)

bit 14-1 **Unimplemented:** Read as ‘0’

bit 0 **PRSSEL:** Channel Priority Scheme Selection bit
1 = Round robin scheme
0 = Fixed priority scheme

6.2 RTSP Operation

The PIC24F Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user to erase blocks of eight rows (512 instructions) at a time and to program one row at a time. It is also possible to program single words.

The 8-row erase blocks and single row write blocks are edge-aligned, from the beginning of program memory on boundaries of 1536 bytes and 192 bytes, respectively.

When data is written to program memory using `TBLWT` instructions, the data is not written directly to memory. Instead, data written using Table Writes is stored in holding latches until the programming sequence is executed.

Any number of `TBLWT` instructions can be executed and a write will be successfully performed. However, 64 `TBLWT` instructions are required to write the full row of memory.

To ensure that no data is corrupted during a write, any unused address should be programmed with FFFFFFh. This is because the holding latches reset to an unknown state, so if the addresses are left in the Reset state, they may overwrite the locations on rows which were not rewritten.

The basic sequence for RTSP programming is to set up a Table Pointer, then do a series of `TBLWT` instructions to load the buffers. Programming is performed by setting the control bits in the NVMCON register.

Data can be loaded in any order and the holding registers can be written to multiple times before performing a write operation. Subsequent writes, however, will wipe out any previous writes.

> **Note:** Writing to a location multiple times without erasing is *not* recommended.

All of the Table Write operations are single-word writes (2 instruction cycles), because only the buffers are written. A programming cycle is required for programming each row.

6.3 JTAG Operation

The PIC24F family supports JTAG boundary scan. Boundary scan can improve the manufacturing process by verifying pin to PCB connectivity.

6.4 Enhanced In-Circuit Serial Programming

Enhanced In-Circuit Serial Programming uses an on-board bootloader, known as the Program Executive (PE), to manage the programming process. Using an SPI data frame format, the Program Executive can erase, program and verify program memory. For more information on Enhanced ICSP, see the device programming specification.

6.5 Control Registers

There are two SFRs used to read and write the program Flash memory: NVMCON and NVMKEY.

The NVMCON register (Register 6-1) controls which blocks are to be erased, which memory type is to be programmed and when the programming cycle starts.

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user must consecutively write 55h and AAh to the NVMKEY register. For more information, refer to **Section 6.6 "Programming Operations"**.

6.6 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. During a programming or erase operation, the processor stalls (waits) until the operation is finished. Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

7.0 RESETS

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"*, **"Reset"** (DS39712). The information in this data sheet supersedes the information in the FRM.

The Reset module combines all Reset sources and controls the device Master Reset Signal, $\overline{\text{SYSRST}}$. The following is a list of device Reset sources:

- POR: Power-on Reset
- $\overline{\text{MCLR}}$: Master Clear Pin Reset
- SWR: `RESET` Instruction
- WDT: Watchdog Timer Reset
- BOR: Brown-out Reset
- CM: Configuration Mismatch Reset
- TRAPR: Trap Conflict Reset
- IOPUWR: Illegal Opcode Reset
- UWR: Uninitialized W Register Reset

A simplified block diagram of the Reset module is shown in Figure 7-1.

Any active source of Reset will make the $\overline{\text{SYSRST}}$ signal active. Many registers associated with the CPU and peripherals are forced to a known Reset state. Most registers are unaffected by a Reset; their status is unknown on POR and unchanged by all other Resets.

Note:	Refer to the specific peripheral or CPU section of this manual for register Reset states.

All types of device Resets will set a corresponding status bit in the RCON register to indicate the type of Reset (see Register 7-1). In addition, Reset events occurring while an extreme power-saving feature is in use (such as VBAT) will set one or more status bits in the RCON2 register (Register 7-2). A POR will clear all bits, except for the BOR and POR (RCON<1:0>) bits, which are set. The user may set or clear any bit at any time during code execution. The RCON bits only serve as status bits. Setting a particular Reset status bit in software will not cause a device Reset to occur.

The RCON register also has other bits associated with the Watchdog Timer and device power-saving states. The function of these bits is discussed in other sections of this data sheet.

Note:	The status bits in the RCON registers should be cleared after they are read so that the next RCON register values after a device Reset will be meaningful.

FIGURE 7-1: RESET SYSTEM BLOCK DIAGRAM

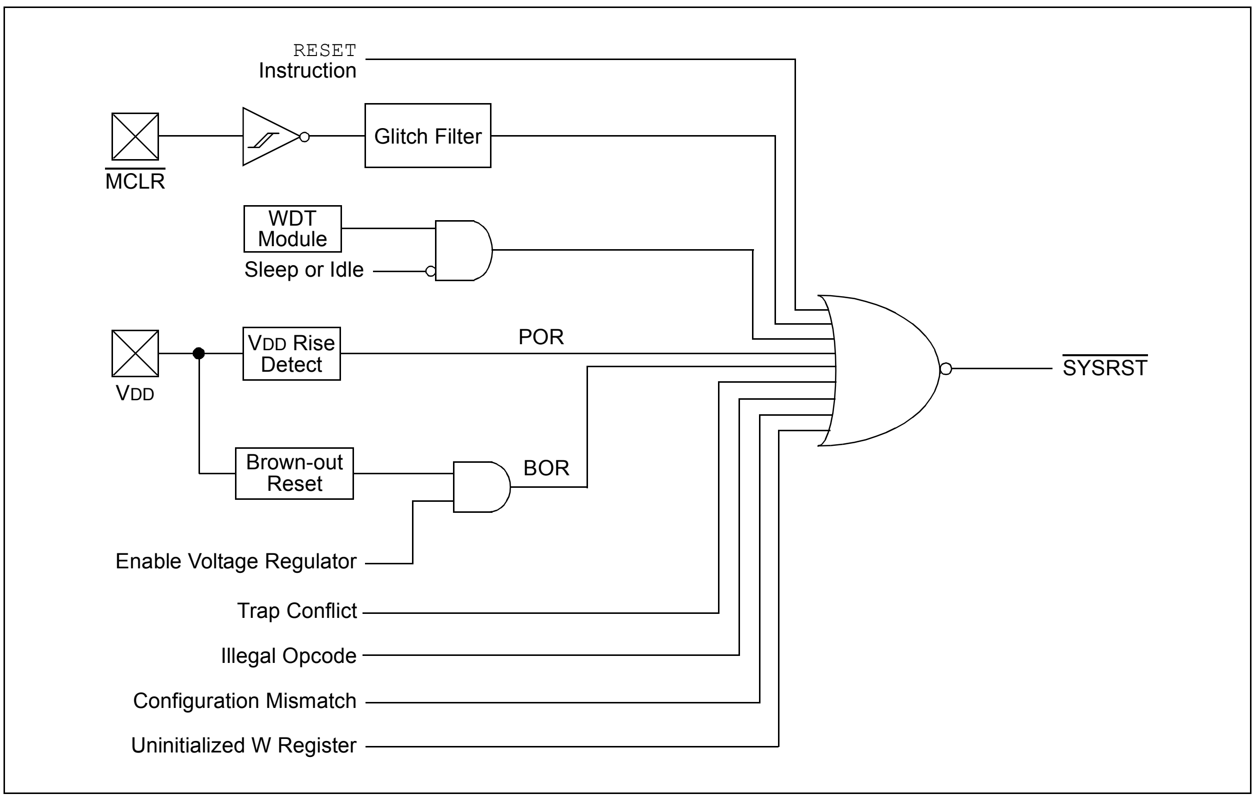


TABLE 8-2: IMPLEMENTED INTERRUPT VECTORS

Interrupt Source	Vector #	IRQ #	IVT Address	AIVT Address	Interrupt Bit Locations		
					Flag	Enable	Priority
ADC1 Interrupt	21	13	00002Eh	00012Eh	IFS0<13>	IEC0<13>	IPC3<6:4>
Comparator Event	26	18	000038h	000138h	IFS1<2>	IEC1<2>	IPC4<10:8>
CRC Generator	75	67	00009Ah	00019Ah	IFS4<3>	IEC4<3>	IPC16<14:12>
CTMU Event	85	77	0000AEh	0001AEh	IFS4<13>	IEC4<13>	IPC19<6:4>
Cryptographic Operation Done	63	55	000082h	000182h	IFS3<7>	IEC3<7>	IPC13<14:12>
Cryptographic Key Store Program Done	64	56	000084h	000184h	IFS3<8>	IEC3<8>	IPC14<2:0>
Cryptographic Buffer Ready	42	34	000058h	000158h	IFS2<2>	IEC2<2>	IPC8<10:8>
Cryptographic Rollover	43	35	00005Ah	00015Ah	IFS2<3>	IEC2<3>	IPC8<14:12>
DMA Channel 0	12	4	00001Ch	00011Ch	IFS0<4>	IEC0<4>	IPC1<2:0>
DMA Channel 1	22	14	000030h	000130h	IFS0<14>	IEC0<14>	IPC3<10:8>
DMA Channel 2	32	24	000044h	000144h	IFS1<8>	IEC1<8>	IPC6<2:0>
DMA Channel 3	44	36	00005Ch	00015Ch	IFS2<4>	IEC2<4>	IPC9<2:0>
DMA Channel 4	54	46	000070h	000170h	IFS2<14>	IEC2<14>	IPC11<10:8>
DMA Channel 5	69	61	00008Eh	00018Eh	IFS3<13>	IEC3<13>	IPC15<6:4>
External Interrupt 0	8	0	000014h	000114h	IFS0<0>	IEC0<0>	IPC0<2:0>
External Interrupt 1	28	20	00003Ch	00013Ch	IFS1<4>	IEC1<4>	IPC5<2:0>
External Interrupt 2	37	29	00004Eh	00014Eh	IFS1<13>	IEC1<13>	IPC7<6:4>
External Interrupt 3	61	53	00007Eh	00017Eh	IFS3<5>	IEC3<5>	IPC13<6:4>
External Interrupt 4	62	54	000080h	000180h	IFS3<6>	IEC3<6>	IPC13<10:8>
FRC Self-Tune	114	106	0000E8h	0001E8h	IFS6<10>	IEC6<10>	IPC26<10:8>
I2C1 Master Event	25	17	000036h	000136h	IFS1<1>	IEC1<1>	IPC4<6:4>
I2C1 Slave Event	24	16	000034h	000134h	IFS1<0>	IEC1<0>	IPC4<2:0>
I2C1 Bus Collision	92	84	0000BC	0001BC	IFS5<4>	IEC5<4>	IPC21<2:0>
I2C2 Master Event	58	50	000078h	000178h	IFS3<2>	IEC3<2>	IPC12<10:8>
I2C2 Slave Event	57	49	000076h	000176h	IFS3<1>	IEC3<1>	IPC12<6:4>
I2C2 Bus Collision.	93	85	0000BE	0001BE	IFS5<5>	IEC5<5>	IPC21<6:4>
Input Capture 1	9	1	000016h	000116h	IFS0<1>	IEC0<1>	IPC0<6:4>
Input Capture 2	13	5	00001Eh	00011Eh	IFS0<5>	IEC0<5>	IPC1<6:4>
Input Capture 3	45	37	00005Eh	00015Eh	IFS2<5>	IEC2<5>	IPC9<6:4>
Input Capture 4	46	38	000060h	000160h	IFS2<6>	IEC2<6>	IPC9<10:8>
Input Capture 5	47	39	000062h	000162h	IFS2<7>	IEC2<7>	IPC9<14:12>
Input Capture 6	48	40	000064h	000164h	IFS2<8>	IEC2<8>	IPC10<2:0>
JTAG	125	117	0000FEh	0001FEh	IFS7<5>	IEC7<5>	IPC29<6:4>
Input Change Notification (ICN)	27	19	00003Ah	00013Ah	IFS1<3>	IEC1<3>	IPC4<14:12>
High/Low-Voltage Detect (HLVD)	80	72	0000A4h	0001A4h	IFS4<8>	IEC4<8>	IPC18<2:0>
Output Compare 1	10	2	000018h	000118h	IFS0<2>	IEC0<2>	IPC0<10:8>
Output Compare 2	14	6	000020h	000120h	IFS0<6>	IEC0<6>	IPC1<10:8>
Output Compare 3	33	25	000046h	000146h	IFS1<9>	IEC1<9>	IPC6<6:4>
Output Compare 4	34	26	000048h	000148h	IFS1<10>	IEC1<10>	IPC6<10:8>
Output Compare 5	49	41	000066h	000166h	IFS2<9>	IEC2<9>	IPC10<6:4>
Output Compare 6	50	42	000068h	000168h	IFS2<10>	IEC2<10>	IPC10<10:8>
Enhanced Parallel Master Port (EPMP)	53	45	00006Eh	00016Eh	IFS2<13>	IEC2<13>	IPC11<6:4>
Real-Time Clock and Calendar (RTCC)	70	62	000090h	000190h	IFS3<14>	IEC3<14>	IPC15<10:8>

REGISTER 8-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1TXIF	SPI1IF	T3IF
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
T2IF	OC2IF	IC2IF	DMA0IF	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 **Unimplemented:** Read as ‘0’

bit 14 **DMA1IF:** DMA Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **AD1IF:** A/D Event 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 **SPI1TXIF:** SPI1 Transmit Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **SPI1IF:** SPI1 General 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 **DMA0IF:** DMA Channel 0 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3 **T1IF:** Timer1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1 (CONTINUED)

bit 2	**CMIF:** Comparator Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 1	**MI2C1IF:** Master I2C1 Event Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 0	**SI2C1IF:** Slave I2C1 Event Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred

REGISTER 8-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIE	U2RXIE	INT2IE[(1)]	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IE[(1)]	CNIE	CMIE	MI2C1IE	SI2C1IE
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 **U2TXIE:** UART2 Transmitter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 14 **U2RXIE:** UART2 Receiver Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **INT2IE:** External Interrupt 2 Enable bit[(1)]
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12 **T5IE:** Timer5 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 11 **T4IE:** Timer4 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 10 **OC4IE:** Output Compare Channel 4 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 9 **OC3IE:** Output Compare Channel 3 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **DMA2IE:** DMA Channel 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **INT1IE:** External Interrupt 1 Enable bit[(1)]
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3 **CNIE:** Input Change Notification Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPn or RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 11-6: RPINR2: PERIPHERAL PIN SELECT INPUT REGISTER 2

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	OCTRIG2R5	OCTRIG2R4	OCTRIG2R3	OCTRIG2R2	OCTRIG2R1	OCTRIG2R0
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
—	—	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0
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 **OCTRIG2R<5:0>:** Assign Output Compare Trigger 2 to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **INT4R<5:0>:** Assign External Interrupt 4 (INT4) to Corresponding RPn or RPIn Pin bits

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

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
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
—	—	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
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 **IC2R<5:0>:** Assign Input Capture 2 (IC2) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **IC1R<5:0>:** Assign Input Capture 1 (IC1) to Corresponding RPn or RPIn Pin bits

15.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the OCxRS and OCxR registers. The OCxRS and OCxR registers can be written to at any time, but the duty cycle value is not latched until a match between PRy and TMRy occurs (i.e., the period is complete). This provides a double buffer for the PWM duty cycle and is essential for glitchless PWM operation.

Some important boundary parameters of the PWM duty cycle include:

- If OCxR, OCxRS and PRy are all loaded with 0000h, the OCx pin will remain low (0% duty cycle).
- If OCxRS is greater than PRy, the pin will remain high (100% duty cycle).

See Example 15-1 for PWM mode timing details. Table 15-1 and Table 15-2 show example PWM frequencies and resolutions for a device operating at 4 MIPS and 10 MIPS, respectively.

EQUATION 15-2: CALCULATION FOR MAXIMUM PWM RESOLUTION(1)

$$\text{Maximum PWM Resolution (bits)} = \frac{\log_{10}\left(\dfrac{F_{CY}}{F_{PWM} \bullet (\text{Timer Prescale Value})}\right)}{\log_{10}} \text{ bits}$$

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

EXAMPLE 15-1: PWM PERIOD AND DUTY CYCLE CALCULATIONS(1)

1. Find the Timer Period register value for a desired PWM frequency of 52.08 kHz, where FOSC = 8 MHz with PLL (32 MHz device clock rate) and a Timer2 prescaler setting of 1:1.

   $T_{CY} = 2 * T_{OSC} = 62.5 \text{ ns}$

   $\text{PWM Period} = 1/\text{PWM Frequency} = 1/52.08 \text{ kHz} = 19.2 \text{ ms}$

   $\text{PWM Period} = (PR2 + 1) \bullet T_{CY} \bullet (\text{Timer2 Prescale Value})$

   $19.2\ \mu s = (PR2 + 1) \bullet 62.5 \text{ ns} \bullet 1$

   $PR2 = 306$

2. Find the maximum resolution of the duty cycle that can be used with a 52.08 kHz frequency and a 32 MHz device clock rate:

   $\text{PWM Resolution} = \log_{10}(F_{CY}/F_{PWM})/\log_{10}2) \text{ bits}$

   $= (\log_{10}(16 \text{ MHz}/52.08 \text{ kHz})/\log_{10}2) \text{ bits}$

   $= 8.3 \text{ bits}$

Note 1: Based on TCY = 2 * TOSC; Doze mode and PLL are disabled.

TABLE 15-1: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 4 MIPS (FCY = 4 MHz)(1)

PWM Frequency	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

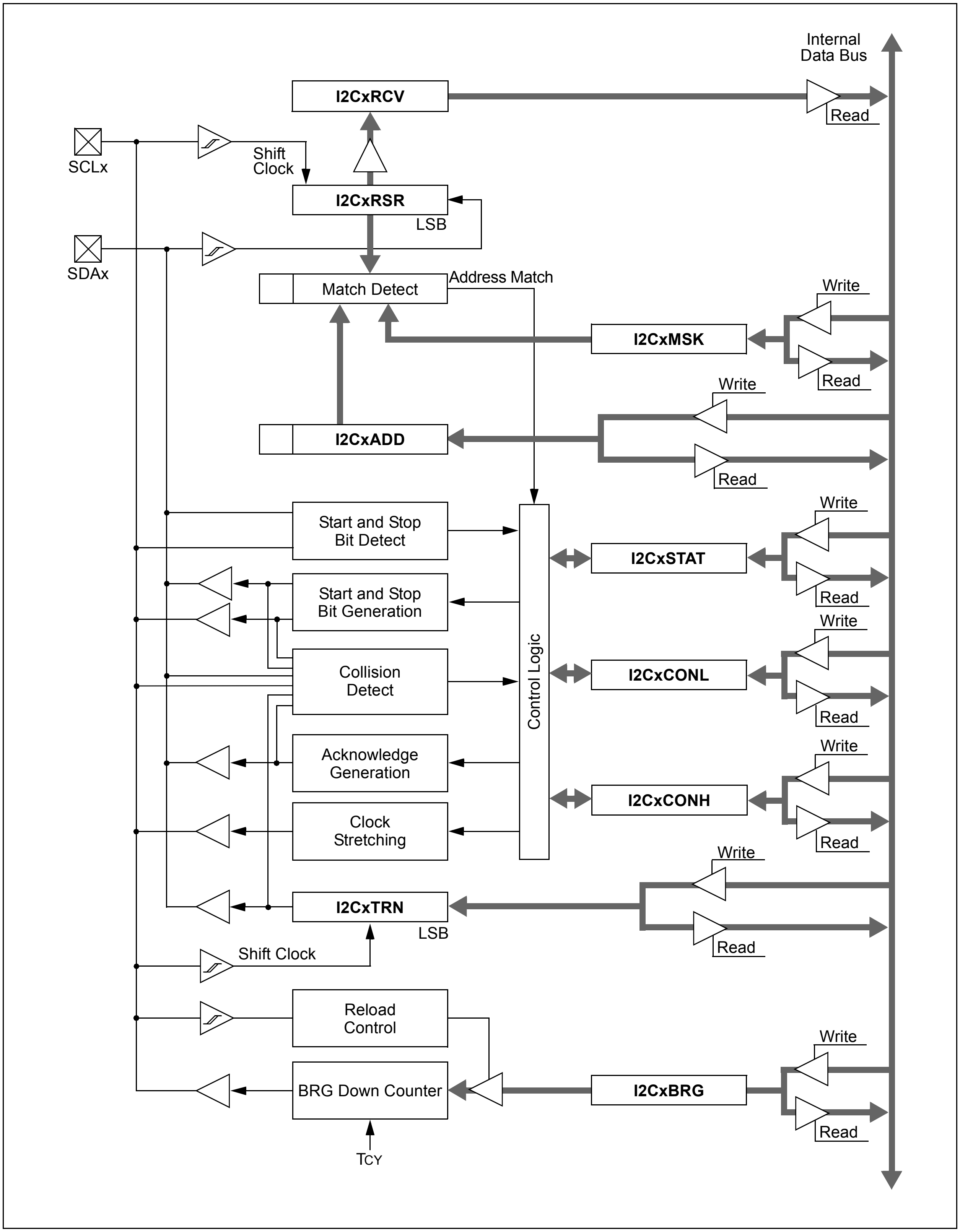
Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

TABLE 15-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 16 MIPS (FCY = 16 MHz)(1)

PWM Frequency	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

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

FIGURE 17-1: I2Cx BLOCK DIAGRAM



18.9 Registers

The UART module consists of the following Special Function Registers (SFRs):

- UxMODE: UARTx Mode Register (Register 18-1)
- UxSTA: UARTx Status and Control Register (Register 18-2)
- UxRXREG: UARTx Receive Register
- UxTXREG: UARTx Transmit Register (Write-Only) (Register 18-3)
- UxADMD: UARTx Address Mask Detect Register (Register 18-4)
- UxBRG: UARTx Baud Rate Register
- UxSCCON: UARTx Smart Card Control Register (Register 18-5)
- UxSCINT: UARTx Smart Card Interrupt Register (Register 18-6)
- UxGTC: UARTx Guard Time Counter Register
- UxWTCL and UxWTCH: UARTx Waiting Time Counter Registers

REGISTER 18-1: UxMODE: UARTx MODE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN(1)	—	USIDL	IREN(2)	RTSMD	—	UEN1	UEN0
bit 15							bit 8

R/W-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7							bit 0

Legend:	HC = Hardware 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 **UARTEN:** UARTx Enable bit(1)
1 = UARTx is enabled; all UARTx pins are controlled by UARTx as defined by UEN<1:0>
0 = UARTx is disabled; all UARTx pins are controlled by port latches; UARTx power consumption is minimal

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **USIDL:** UARTx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **IREN:** IrDA® Encoder and Decoder Enable bit(2)
1 = IrDA encoder and decoder are enabled
0 = IrDA encoder and decoder are disabled

bit 11 **RTSMD:** Mode Selection for UxRTS Pin bit
1 = UxRTS pin is in Simplex mode
0 = UxRTS pin is in Flow Control mode

bit 10 **Unimplemented:** Read as ‘0’

bit 9-8 **UEN<1:0>:** UARTx Enable bits
11 = UxTX, UxRX and BCLKx pins are enabled and used; UxCTS pin is controlled by port latches
10 = UxTX, UxRX, UxCTS and UxRTS pins are enabled and used
01 = UxTX, UxRX and UxRTS pins are enabled and used; UxCTS pin is controlled by port latches
00 = UxTX and UxRX pins are enabled and used; UxCTS and UxRTS/BCLKx pins are controlled by port latches

bit 7 **WAKE:** Wake-up on Start Bit Detect During Sleep Mode Enable bit
1 = UARTx continues to sample the UxRX pin; interrupt is generated on the falling edge, bit is cleared in hardware on the following rising edge
0 = No wake-up is enabled

Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 “Peripheral Pin Select (PPS)”**.
2: This feature is only available for the 16x BRG mode (BRGH = 0).

24.2 Extended DMA Operations

In addition to the standard features available on all 12-bit A/D Converters, PIC24FJ128GA204 family devices implement a limited extension of DMA functionality. This extension adds features that work with the device's DMA Controller to expand the A/D module's data storage abilities beyond the module's built-in buffer.

The Extended DMA functionality is controlled by the DMAEN bit (AD1CON1<11>); setting this bit enables the functionality. The DMABM bit (AD1CON1<12>) configures how the DMA feature operates.

24.2.1 EXTENDED BUFFER MODE

Extended Buffer mode (DMABM = `1`) is useful for storing the results of channels. It can also be used to store the conversion results on any A/D channel in any implemented address in data RAM.

In Extended Buffer mode, all data from the A/D Buffer register, and channels above 26, is mapped into data RAM. Conversion data is written to a destination specified by the DMA Controller, specifically by the DMADSTn register. This allows users to read the conversion results of channels above 26, which do not have their own memory-mapped A/D buffer locations, from data memory.

When using Extended Buffer mode, always set the BUFREGEN bit to disable FIFO operation. In addition, disable the Split Buffer mode by clearing the BUFM bit.

24.2.2 PIA MODE

When DMABM = `0`, the A/D module is configured to function with the DMA Controller for Peripheral Indirect Addressing (PIA) mode operations. In this mode, the A/D module generates an 11-bit Indirect Address (IA). This is ORed with the destination address in the DMA Controller to define where the A/D conversion data will be stored.

In PIA mode, the buffer space is created as a series of contiguous smaller buffers, one per analog channel. The size of the channel buffer determines how many analog channels can be accommodated. The size of the buffer is selected by the DMABL<2:0> bits (AD1CON4<2:0>). The size options range from a single word per buffer to 128 words. Each channel is allocated a buffer of this size, regardless of whether or not the channel will actually have conversion data.

The IA is created by combining the base address within a channel buffer with three to five bits (depending on the buffer size) to identify the channel. The base address ranges from zero to seven bits wide, depending on the buffer size. The address is right-padded with a '`0`' in order to maintain address alignment in the Data Space. The concatenated channel and base address bits are then left-padded with zeros, as necessary, to complete the 11-bit IA.

The IA is configured to auto-increment during write operations by using the SMPIx bits (AD1CON2<6:2>).

As with PIA operations for any DMA-enabled module, the base destination address in the DMADSTn register must be masked properly to accommodate the IA. Table 24-1 shows how complete addresses are formed. Note that the address masking varies for each buffer size option. Because of masking requirements, some address ranges may not be available for certain buffer sizes. Users should verify that the DMA base address is compatible with the buffer size selected.

Figure 24-2 shows how the parts of the address define the buffer locations in data memory. In this case, the module "allocates" 256 bytes of data RAM (1000h to 1100h) for 32 buffers of four words each. However, this is not a hard allocation and nothing prevents these locations from being used for other purposes. For example, in the current case, if Analog Channels 1, 3 and 8 are being sampled and converted, conversion data will only be written to the channel buffers, starting at 1008h, 1018h and 1040h. The holes in the PIA buffer space can be used for any other purpose. It is the user's responsibility to keep track of buffer locations and prevent data overwrites.

24.3 A/D Operation with V_{BAT}

One of the A/D channels is connected to the V_{BAT} pin to monitor the V_{BAT} voltage. This allows monitoring the V_{BAT} pin voltage (battery voltage) with no external connection. The voltage measured, using the A/D V_{BAT} monitor, is $V_{BAT}/2$. The voltage can be calculated by reading A/D = $((V_{BAT}/2)/V_{DD}) * 1024$ for 10-bit A/D and $((V_{BAT}/2)/V_{DD}) * 4096$ for 12 bit A/D.

When using the V_{BAT} A/D monitor:

- Connect the A/D channel to ground to discharge the sample capacitor.
- Because of the high-impedance of V_{BAT}, select higher sampling time to get an accurate reading.

Since the V_{BAT} pin is connected to the A/D during sampling, to prolong the V_{BAT} battery life, the recommendation is to only select the V_{BAT} channel when needed.

REGISTER 24-1: AD1CON1: A/D CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	DMABM[(1)]	DMAEN	MODE12	FORM1	FORM0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0, HSC	R/C-0, HSC
SSRC3	SSRC2	SSRC1	SSRC0	—	ASAM	SAMP	DONE
bit 7							bit 0

Legend:	C = Clearable bit	U = Unimplemented bit, read as ‘0’	
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-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 = 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 **DMABM:** Extended DMA Buffer Mode Select bit[(1)]
1 = Extended Buffer mode: Buffer address is defined by the DMADSTn register
0 = PIA mode: Buffer addresses are defined by the DMA Controller and AD1CON4<2:0>

bit 11 **DMAEN:** Extended DMA/Buffer Enable bit
1 = Extended DMA and buffer features are enabled
0 = Extended features are disabled

bit 10 **MODE12:** 12-Bit Operation Mode bit
1 = 12-bit A/D operation
0 = 10-bit A/D operation

bit 9-8 **FORM<1:0>:** Data Output Format bits (see formats following)
11 = Fractional result, signed, left justified
10 = Absolute fractional result, unsigned, left justified
01 = Decimal result, signed, right justified
00 = Absolute decimal result, unsigned, right justified

bit 7-4 **SSRC<3:0>:** Sample Clock Source Select bits
1xxx = Unimplemented, do not use
0111 = Internal counter ends sampling and starts conversion (auto-convert); do not use in Auto-Scan mode
0110 = Unimplemented
0101 = TMR1
0100 = CTMU
0011 = TMR5
0010 = TMR3
0001 = INT0
0000 = The SAMP bit must be cleared by software to start conversion

bit 3 **Unimplemented:** Read as ‘0’

bit 2 **ASAM:** A/D Sample Auto-Start bit
1 = Sampling begins immediately after last conversion; SAMP bit is auto-set
0 = Sampling begins when SAMP bit is manually set

Note 1: This bit is only available when Extended DMA/Buffer features are available (DMAEN = 1).

NOTES:

29.4.3 CONFIGURATION REGISTER PROTECTION

The Configuration registers are protected against inadvertent or unwanted changes or reads in two ways. The primary protection method is the same as that of the RP registers – shadow registers contain a complimentary value which is constantly compared with the actual value.

To safeguard against unpredictable events, Configuration bit changes resulting from individual cell-level disruptions (such as ESD events) will cause a parity error and trigger a device Reset.

The data for the Configuration registers is derived from the Flash Configuration Words in program memory. When the GCP bit is set, the source data for device configuration is also protected as a consequence. Even if General Segment protection is not enabled, the device configuration can be protected by using the appropriate Code Segment protection setting.

29.5 JTAG Interface

PIC24FJ128GA204 family devices implement a JTAG interface, which supports boundary scan device testing and programming.

29.6 In-Circuit Serial Programming

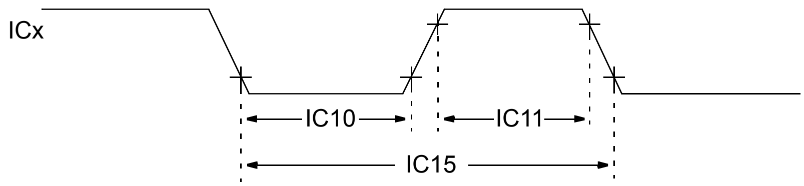
PIC24FJ128GA204 family microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock (PGECx) and data (PGEDx), and three other lines for power (V_{DD}), ground (V_{SS}) and $\overline{MCLR}$. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

29.7 In-Circuit Debugger

When MPLAB® ICD 3 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pins.

To use the in-circuit debugger function of the device, the design must implement ICSP connections to $\overline{MCLR}$, V_{DD}, V_{SS} and the PGECx/PGEDx pin pair, designated by the ICSx Configuration bits. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

FIGURE 32-10: INPUT CAPTURE x (ICx) TIMING CHARACTERISTICS



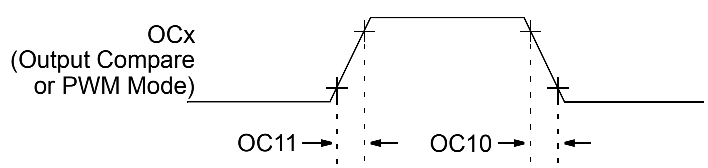
Note: Refer to Figure 32-2 for load conditions.

TABLE 32-32: INPUT CAPTURE x 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.	**Symbol**	**Characteristic[1]**		**Min**	**Max**	**Units**	**Conditions**
IC10	TccL	ICx Input Low Time	No Prescaler	0.5 TCY + 20	—	ns	
			With Prescaler	10	—	ns	
IC11	TccH	ICx Input High Time	No Prescaler	0.5 TCY + 20	—	ns	
			With Prescaler	10	—	ns	
IC15	TccP	ICx Input Period		(TCY + 40)/N	—	ns	N = Prescale Value (1, 4, 16)

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

FIGURE 32-11: OUTPUT COMPARE x MODULE (OCx) TIMING CHARACTERISTICS



Note: Refer to Figure 32-2 for load conditions.

TABLE 32-33: OUTPUT COMPARE x MODULE 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.	**Symbol**	**Characteristic[1]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
OC10	TccF	OCx Output Fall Time	—	—	—	ns	See Parameter DO32
OC11	TccR	OCx Output Rise Time	—	—	—	ns	See Parameter DO31

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

NOTES: