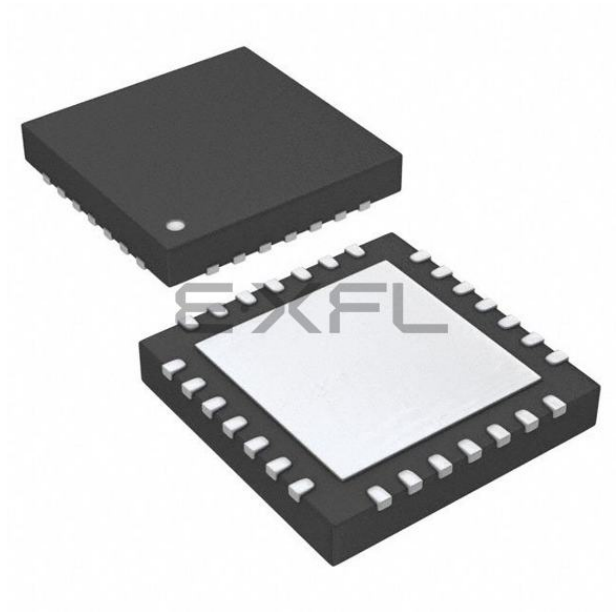
Microchip Technology - PIC24FV08KM202-I/ML Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

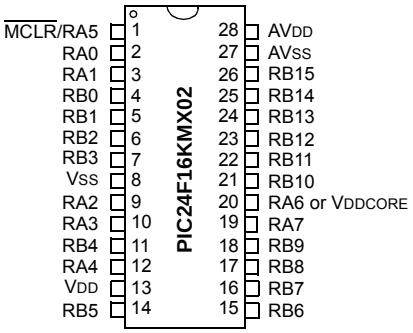
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	23
Program Memory Size	8KB (2.75K x 24)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5V
Data Converters	A/D 19x10b/12b; D/A 2x8b
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/pic24fv08km202-i-ml

Pin Diagrams (Continued)

28-Pin SPDIP/SSOP/SOIC



Pin	Pin Features	
	PIC24FXXKMX02	PIC24FVXXKMX02
1	MCLR/VPP/RA5	
2	CVREF+/VREF+/ /AN0/ /CN2/RA0	
3	CVREF-/VREF-/AN1/CN3/RA1	CVREF-/VREF-/AN1/RA1
4	PGED1/AN2/CTCMP/ULPWU/C1IND/ / / /CN4/RB0	
5	PGEC1/ / /AN3/C1INC/ / /CTED12/CN5/RB1	
6	/ /AN4/C1INB/ / /U1RX/TCKIB/CTED13/CN6/RB2	
7	/AN5/C1INA/ / /CN7/RB3	
8	VSS	
9	OSCI/CLKI/AN13/CN30/RA2	
10	OSCO/CLKO/AN14/CN29/RA3	
11	SOSCI/AN15/ / /CN1/RB4	
12	SOSCO/SCLKI/AN16/PWRLCLK/ /CN0/RA4	
13	VDD	
14	PGED3/AN17/ASDA1/ / /OC1E/CLCINA/CN27/RB5	
15	PGEC3/AN18/ASCL1/ / /OC1F/CLCINB/CN24/RB6	
16	AN19/U1TX/INT0/CN23/RB7	AN19/U1TX/ / /INT0/CN23/RB7
17	AN20/SCL1/U1CTS/C3OUT/OC1B/CTED10/CN22/RB8	
18	AN21/SDA1/T1CK/U1RTS/U1BCLK/IC2/ /CLC1O/CTED4/CN21/RB9	
19	/IC1/ / /CTED3/CN9/RA7	
20	/OC1A/CTED1/INT2/CN8/RA6	VCAP OR VDDCORE
21	PGED2/SDI1/ /OC1C/CTED11/CN16/RB10	
22	PGEC2/SCK1/OC2A/CTED9/CN15/RB11	
23	/AN12/HLVDIN/ / / /CTED2/CN14/RB12	/AN12/HLVDIN/ / / /CTED2/INT2/CN14/RB12
24	/ /AN11/SDO1/OCFB/ /OC1D/CTPLS/CN13/RB13	
25	/CVREF/ / /AN10/ / /C1OUT/OCFA/CTED5/INT1/CN12/RB14	
26	/ /AN9/ /REFO/SS1/TCKIA/CTED6/CN11/RB15	
27	VSS/AVSS	
28	VDD/AVDD	

Legend: Values in indicate pin function differences between PIC24F(V)XXKM202 and PIC24F(V)XXKM102 devices.

TABLE 1-5: PIC24FV16KM204 FAMILY PINOUT DESCRIPTION (CONTINUED)

Function	F					FV					I/O	Buffer	Description
	Pin Number					Pin Number							
	20-Pin PDIP/ SSOP/ SOIC	28-Pin PDIP/ SSOP/ SOIC	28-Pin QFN	44-Pin QFN/ TQFP	48-Pin UQFN	20-Pin PDIP/ SSOP/ SOIC	28-Pin PDIP/ SSOP/ SOIC	28-Pin QFN	44-Pin QFN/ TQFP	48-Pin UQFN			
RB9	13	18	15	1	1	13	18	15	1	1	I/O	ST	PORTB Pins
RB10	—	21	18	8	9	—	21	18	8	9	I/O	ST	PORTB Pins
RB11	—	22	19	9	10	—	22	19	9	10	I/O	ST	PORTB Pins
RB12	15	23	20	10	11	15	23	20	10	11	I/O	ST	PORTB Pins
RB13	16	24	21	11	12	16	24	21	11	12	I/O	ST	PORTB Pins
RB14	17	25	22	14	15	17	25	22	14	15	I/O	ST	PORTB Pins
RB15	18	26	23	15	16	18	26	23	15	16	I/O	ST	PORTB Pins
RC0	—	—	—	25	27	—	—	—	25	27	I/O	ST	PORTC Pins
RC1	—	—	—	26	28	—	—	—	26	28	I/O	ST	PORTC Pins
RC2	—	—	—	27	29	—	—	—	27	29	I/O	ST	PORTC Pins
RC3	—	—	—	36	39	—	—	—	36	39	I/O	ST	PORTC Pins
RC4	—	—	—	37	40	—	—	—	37	40	I/O	ST	PORTC Pins
RC5	—	—	—	38	41	—	—	—	38	41	I/O	ST	PORTC Pins
RC6	—	—	—	2	2	—	—	—	2	2	I/O	ST	PORTC Pins
RC7	—	—	—	3	3	—	—	—	3	3	I/O	ST	PORTC Pins
RC8	—	—	—	4	4	—	—	—	4	4	I/O	ST	PORTC Pins
RC9	—	—	—	5	5	—	—	—	5	5	I/O	ST	PORTC Pins
REFO	18	26	23	15	16	18	26	23	15	16	O	—	Reference Clock Output
RTCC	—	25	22	14	15	—	25	22	14	15	O	—	Real-Time Clock/Calendar Output
SCK1	15	22	19	9	10	15	22	19	9	10	I/O	ST	MSSP1 SPI Clock
SDI1	17	21	18	8	9	17	21	18	8	9	I	ST	MSSP1 SPI Data Input
SDO1	16	24	21	11	12	16	24	21	11	12	O	—	MSSP1 SPI Data Output
$\overline{SS1}$	18	26	23	15	16	18	26	23	15	16	I	ST	MSSP1 SPI Slave Select Input
SCK2	—	14	11	38	41	—	14	11	38	41	I/O	ST	MSSP2 SPI Clock
SDI2	—	19	16	36	39	—	19	16	36	39	I	ST	MSSP2 SPI Data Input
SDO2	—	15	12	37	40	—	15	12	37	40	O	—	MSSP2 SPI Data Output
$\overline{SS2}$	—	23	20	35	38	—	23	20	35	38	I	ST	MSSP2 SPI Slave Select Input

Legend: ANA = Analog level input/output, ST = Schmitt Trigger input buffer, I^2C™ = I^2C/SMBus input buffer

TABLE 4-11: SCCP4 REGISTER MAP

File Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CCP4CON1L(1)	1ACh	CCPON	—	CCPSIDL	r	TMRSYNC	CLKSEL2	CLKSEL1	CLKSEL0	TMRPS1	TMRPS0	T32	CCSEL	MOD3	MOD2	MOD1	MOD0	0000
CCP4CON1H(1)	1AEh	OPSSRC	RTRGEN	—	—	IOPS3	IOPS2	IOPS1	IOPS0	TRIGEN	ONESHOT	ALTSYNC	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	0000
CCP4CON2L(1)	1B0h	PWMRSEN	ASDGM	—	SSDG	—	—	—	—	ASDG7	ASDG6	ASDG5	ASDG4	ASDG3	ASDG2	ASDG1	ASDG0	0000
CCP4CON2H(1)	1B2h	OENSYNC	—	—	—	—	—	—	OCAEN	ICGSM1	ICGSM0	—	AUXOUT1	AUXOUT0	ICSEL2	ICSEL1	ICSEL0	0100
CCP4CON3H(1)	1B6h	OETRIG	OSCNT2	OSCNT1	OSCNT0	—	—	—	—	—	—	POLACE	—	PSSACE1	PSSACE0	—	—	0000
CCP4STATL(1)	1B8h	—	—	—	—	—	—	—	—	CCPTRIG	TRSET	TRCLR	ASEVT	SCEVT	ICDIS	ICOV	ICBNE	0000
CCP4TMRL(1)	1BCh	SCCP4 Time Base Register Low Word																0000
CCP4TMRH(1)	1BEh	SCCP4 Time Base Register High Word																0000
CCP4PRL(1)	1C0h	SCCP4 Time Base Period Register Low Word																FFFF
CCP4PRH(1)	1C2h	SCCP4 Time Base Period Register High Word																FFFF
CCP4RAL(1)	1C4h	Output Compare 4 Data Word A																0000
CCP4RBL(1)	1C8h	Output Compare 4 Data Word B																0000
CCP4BUFL(1)	1CCh	Input Capture 4 Data Buffer Low Word																0000
CCP4BUFH(1)	1CEh	Input Capture 4 Data Buffer High Word																0000

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition, r = reserved.
Note 1: These registers are available only on PIC24F(V)16KM2XX devices.

6.4.1 ERASE DATA EEPROM

The data EEPROM can be fully erased, or can be partially erased, at three different sizes: one word, four words or eight words. The bits, NVMOP<1:0> (NVMCON<1:0>), decide the number of words to be erased. To erase partially from the data EEPROM, the following sequence must be followed:

1. Configure NVMCON to erase the required number of words: one, four or eight.
2. Load TBLPAG and WREG with the EEPROM address to be erased.
3. Clear the NVMIF status bit and enable the NVM interrupt (optional).
4. Write the key sequence to NVMKEY.
5. Set the WR bit to begin the erase cycle.
6. Either poll the WR bit or wait for the NVM interrupt (NVMIF is set).

A typical erase sequence is provided in Example 6-2. This example shows how to do a one-word erase. Similarly, a four-word erase and an eight-word erase can be done. This example uses C library procedures to manage the Table Pointer (`builtin_tblpage` and `builtin_tbloffset`) and the Erase Page Pointer (`builtin_tblwtl`). The memory unlock sequence (`builtin_write_NVM`) also sets the WR bit to initiate the operation and returns control when complete.

EXAMPLE 6-2: SINGLE-WORD ERASE

```
int __attribute__ ((space(eedata))) eeData = 0x1234;
/*-----------------------------------------------------------------------------------------------
The variable eeData must be a Global variable declared outside of any method

the code following this comment can be written inside the method that will execute the erase
-----------------------------------------------------------------------------------------------
*/
    unsigned int offset;

    // Set up NVMCON to erase one word of data EEPROM
    NVMCON = 0x4058;

    // Set up a pointer to the EEPROM location to be erased
    TBLPAG = __builtin_tblpage(&eeData);            // Initialize EE Data page pointer
    offset = __builtin_tbloffset(&eeData);          // Initizlize lower word of address
    __builtin_tblwtl(offset, 0);                    // Write EEPROM data to write latch

    asm volatile ("disi #5");                       // Disable Interrupts For 5 Instructions
    __builtin_write_NVM();                          // Issue Unlock Sequence & Start Write Cycle
    while(NVMCONbits.WR=1);                         // Optional: Poll WR bit to wait for
                                                    // write sequence to complete
```

REGISTER 8-2: CORCON: CPU CONTROL REGISTER

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

U-0	U-0	U-0	U-0	R/C-0, HSC	R/W-0	U-0	U-0
—	—	—	—	IPL3(2)	PSV(1)	—	—
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 15-4 **Unimplemented:** Read as '0'

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit(2)
1 = CPU Interrupt Priority Level is greater than 7
0 = CPU Interrupt Priority Level is 7 or less

bit 1-0 **Unimplemented:** Read as '0'

Note 1: See Register 3-2 for the description of this bit, which is not dedicated to interrupt control functions.
2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.

Note: Bit 2 is described in **Section 3.0 "CPU"**.

REGISTER 8-16: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
DAC2IE	DAC1IE	CTMUIE	—	—	—	—	HLVDIE
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	U2ERIE	U1ERIE	—
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 **DAC2IE:** Digital-to-Analog Converter 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 14 **DAC1IE:** Digital-to-Analog Converter 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **CTMUIE:** CTMU Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

bit 8 **HLVDIE:** High/Low-Voltage Detect Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **U2ERIE:** UART2 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **U1ERIE:** UART1 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

REGISTER 8-29: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0
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 15-11 **Unimplemented:** Read as ‘0’

bit 10-8 **RTCIP<2:0>:** Real-Time Clock and Calendar Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

REGISTER 8-35: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

R-0	U-0	R/W-0	U-0	R-0	R-0	R-0	R-0
CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0
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 **CPUIRQ:** Interrupt Request from Interrupt Controller CPU bit
1 = An interrupt request has occurred but has not yet been Acknowledged by the CPU (this will happen when the CPU priority is higher than the interrupt priority)
0 = No interrupt request is left unacknowledged

bit 14 **Unimplemented:** Read as '0'

bit 13 **VHOLD:** Vector Hold bit
Allows Vector Number Capture and Changes which Interrupt is Stored in the VECNUM<6:0> bits:
1 = VECNUM<6:0> will contain the value of the highest priority pending interrupt, instead of the current interrupt
0 = VECNUM<6:0> will contain the value of the last Acknowledged interrupt (last interrupt that has occurred with higher priority than the CPU, even if other interrupts are pending)

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

bit 11-8 **ILR<3:0>:** New CPU Interrupt Priority Level bits
1111 = CPU Interrupt Priority Level is 15
•
•
•
0001 = CPU Interrupt Priority Level is 1
0000 = CPU Interrupt Priority Level is 0

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

bit 6-0 **VECNUM<6:0>:** Vector Number of Pending Interrupt bits
0111111 = Interrupt vector pending is Number 135
•
•
•
0000001 = Interrupt vector pending is Number 9
0000000 = Interrupt vector pending is Number 8

13.3 Output Compare Mode

Output Compare mode compares the Timer register value with the value of one or two Compare registers, depending on its mode of operation. The Output Compare x module on compare match events has the ability to generate a single output transition or a train of output pulses. Like most PIC® MCU peripherals, the Output Compare x module can also generate interrupts on a compare match event.

Table 13-3 shows the various modes available in Output Compare modes.

TABLE 13-3: OUTPUT COMPARE/PWM MODES

MOD<3:0> (CCPxCON1L<3:0>)	T32 (CCPxCON1L<5>)	Operating Mode	
0001	0	Output High on Compare (16-bit)	Single Edge Mode
0001	1	Output High on Compare (32-bit)	
0010	0	Output Low on Compare (16-bit)	
0010	1	Output Low on Compare (32-bit)	
0011	0	Output Toggle on Compare (16-bit)	
0011	1	Output Toggle on Compare (32-bit)	
0100	0	Dual Edge Compare (16-bit)	Dual Edge Mode
0101	0	Dual Edge Compare (16-bit buffered)	PWM Mode
0110	0	Center-Aligned Pulse (16-bit buffered)	Center PWM
0111	0	Variable Frequency Pulse (16-bit)	
0111	1	Variable Frequency Pulse (32-bit)	

FIGURE 13-5: OUTPUT COMPARE x BLOCK DIAGRAM

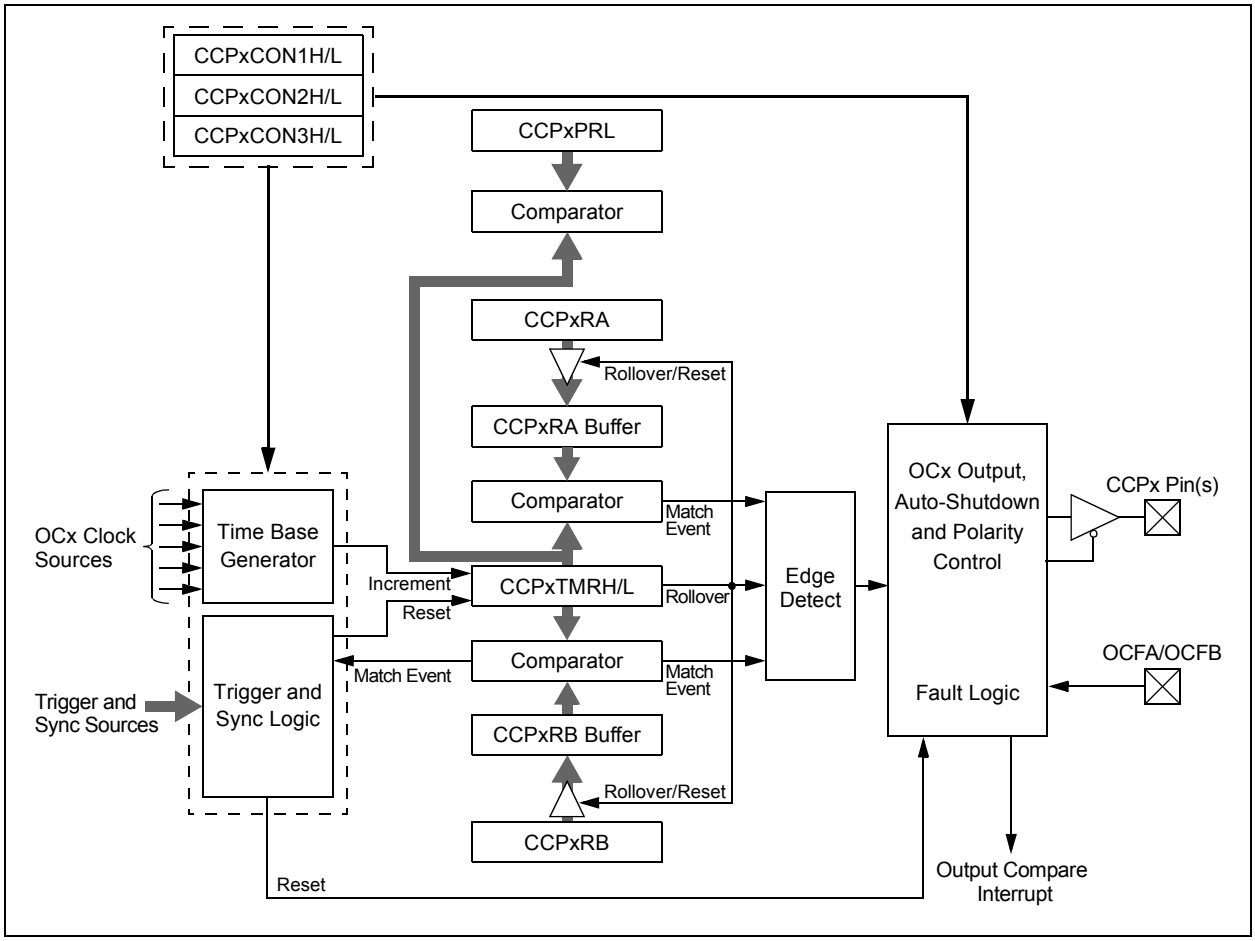
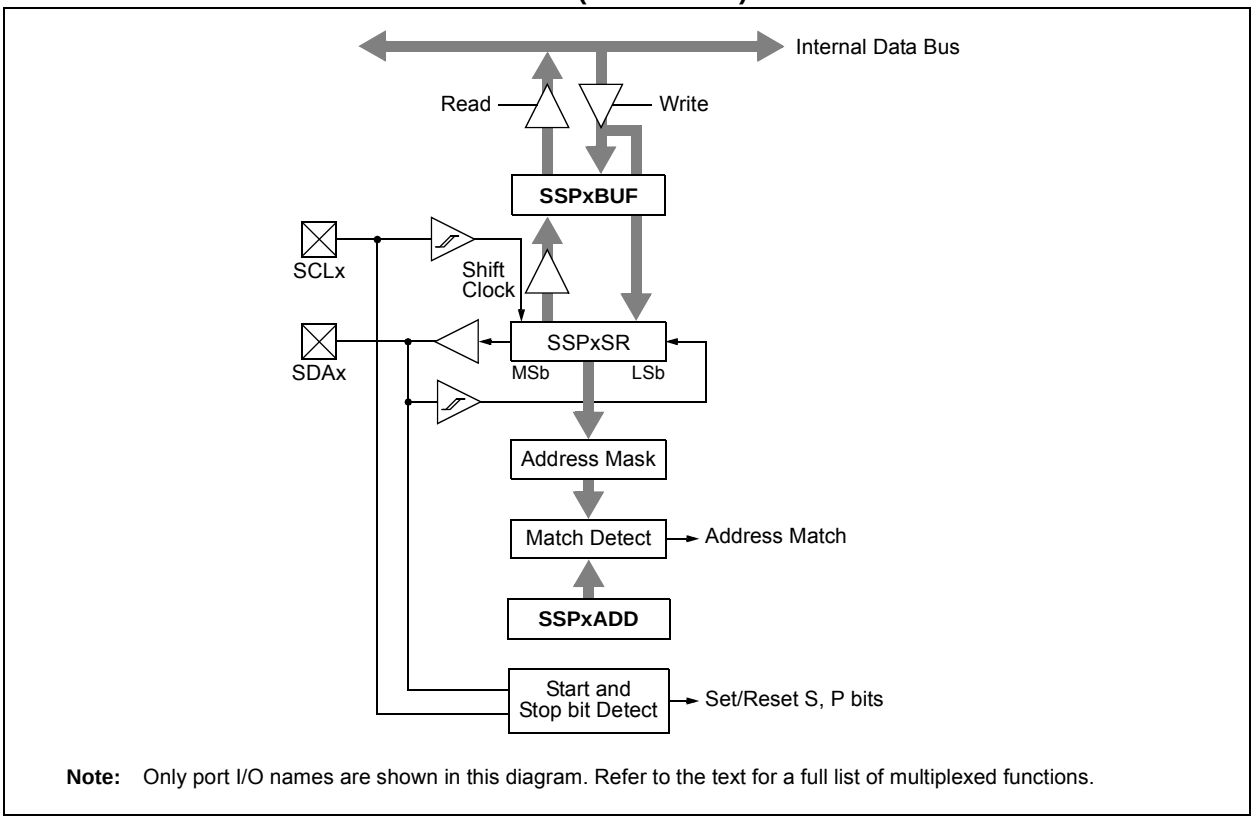
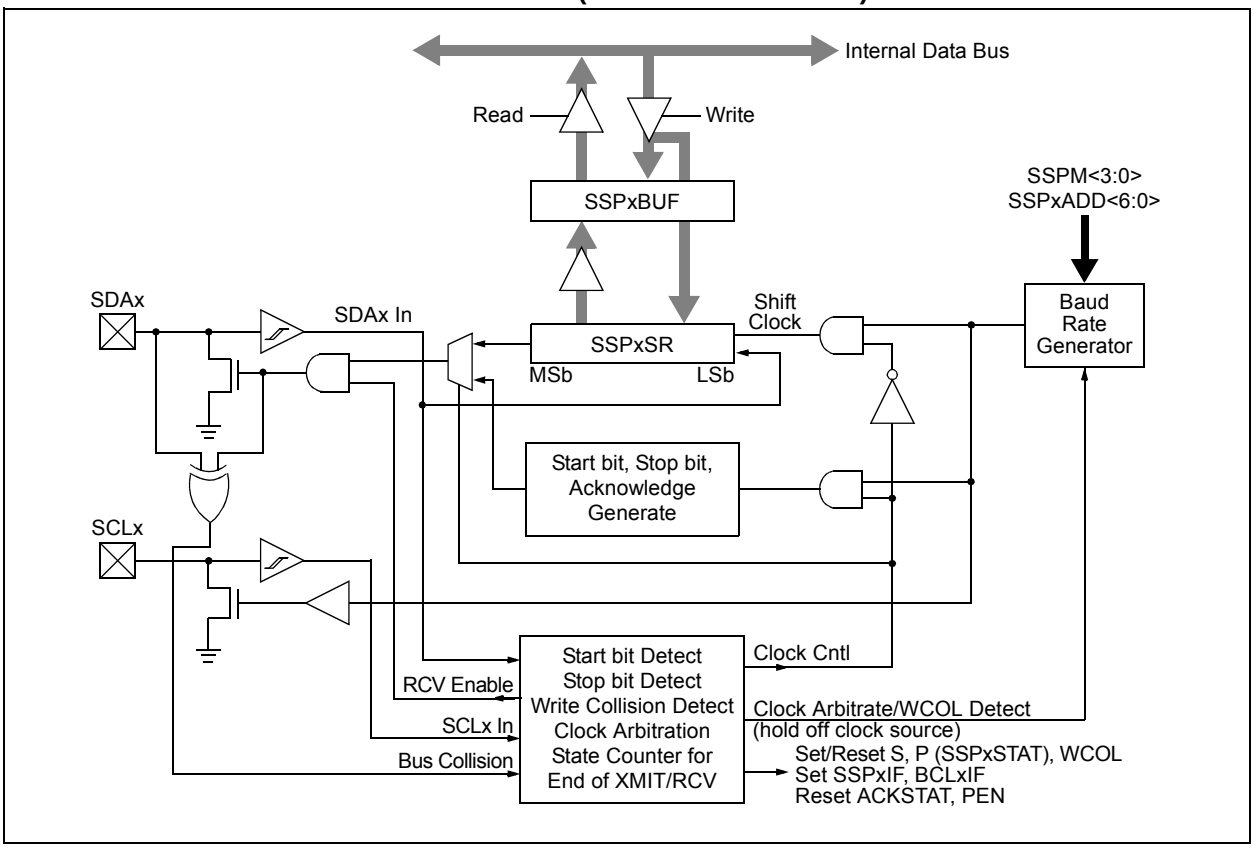


FIGURE 14-3: MSSPx BLOCK DIAGRAM (I^2C™ MODE)



Note: Only port I/O names are shown in this diagram. Refer to the text for a full list of multiplexed functions.

FIGURE 14-4: MSSPx BLOCK DIAGRAM (I^2C™ MASTER MODE)



REGISTER 14-2: SSPxSTAT: MSSPx STATUS REGISTER (I^2C™ MODE)

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

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P(1)	S(1)	R/$\overline{W}$	UA	BF
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **SMP:** Slew Rate Control bit
In Master or Slave mode:
1 = Slew rate control is disabled for Standard Speed mode (100 kHz and 1 MHz)
0 = Slew rate control is enabled for High-Speed mode (400 kHz)

bit 6 **CKE:** SMBus Select bit
In Master or Slave mode:
1 = Enables SMBus-specific inputs
0 = Disables SMBus-specific inputs

bit 5 **D/$\overline{A}$:** Data/$\overline{Address}$ bit
In Master mode:
Reserved.
In Slave mode:
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P:** Stop bit(1)
1 = Indicates that a Stop bit has been detected last
0 = Stop bit was not detected last

bit 3 **S:** Start bit(1)
1 = Indicates that a Start bit has been detected last
0 = Start bit was not detected last

bit 2 **R/$\overline{W}$:** Read/$\overline{Write}$ Information bit
In Slave mode:(2)
1 = Read
0 = Write
In Master mode:(3)
1 = Transmit is in progress
0 = Transmit is not in progress

bit 1 **UA:** Update Address bit (10-Bit Slave mode only)
1 = Indicates that the user needs to update the address in the SSPxADD register
0 = Address does not need to be updated

Note 1: This bit is cleared on Reset and when SSPEN is cleared.

2: This bit holds the R/$\overline{W}$ bit information following the last address match. This bit is only valid from the address match to the next Start bit, Stop bit or not $\overline{ACK}$ bit.

3: ORing this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSPx is in Active mode.

15.2 Transmitting in 8-Bit Data Mode

1. Set up the UARTx:
   a) Write the appropriate values for data, parity and Stop bits.
   b) Write the appropriate baud rate value to the UxBRG register.
   c) Set up transmit and receive interrupt enable and priority bits.
2. Enable the UARTx.
3. Set the UTXEN bit (causes a transmit interrupt, two cycles after being set).
4. Write the data byte to the lower byte of the UxTXREG word. The value will be immediately transferred to the Transmit Shift Register (TSR) and the serial bit stream will start shifting out with the next rising edge of the baud clock.
5. Alternately, the data byte may be transferred while UTXEN = 0, and then, the user may set UTXEN. This will cause the serial bit stream to begin immediately, because the baud clock will start from a cleared state.
6. A transmit interrupt will be generated as per interrupt control bit, UTXISELx.

15.3 Transmitting in 9-Bit Data Mode

1. Set up the UARTx (as described in **Section 15.2 "Transmitting in 8-Bit Data Mode"**).
2. Enable the UARTx.
3. Set the UTXEN bit (causes a transmit interrupt, two cycles after being set).
4. Write UxTXREG as a 16-bit value only.
5. A word write to UxTXREG triggers the transfer of the 9-bit data to the TSR. The serial bit stream will start shifting out with the first rising edge of the baud clock.
6. A transmit interrupt will be generated as per the setting of control bit, UTXISELx.

15.4 Break and Sync Transmit Sequence

The following sequence will send a message frame header, made up of a Break, followed by an Auto-Baud Sync byte.

1. Configure the UARTx for the desired mode.
2. Set UTXEN and UTXBRK – this sets up the Break character.
3. Load the UxTXREG with a dummy character to initiate transmission (value is ignored).
4. Write '55h' to UxTXREG – loads the Sync character into the transmit FIFO.
5. After the Break has been sent, the UTXBRK bit is reset by hardware. The Sync character now transmits.

15.5 Receiving in 8-Bit or 9-Bit Data Mode

1. Set up the UARTx (as described in **Section 15.2 "Transmitting in 8-Bit Data Mode"**).
2. Enable the UARTx.
3. A receive interrupt will be generated when one or more data characters have been received, as per interrupt control bit, URXISELx.
4. Read the OERR bit to determine if an overrun error has occurred. The OERR bit must be reset in software.
5. Read UxRXREG.

The act of reading the UxRXREG character will move the next character to the top of the receive FIFO, including a new set of PERR and FERR values.

15.6 Operation of $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Control Pins

UARTx Clear-To-Send ($\overline{\text{UxCTS}}$) and Request-To-Send ($\overline{\text{UxRTS}}$) are the two hardware controlled pins that are associated with the UARTx module. These two pins allow the UARTx to operate in Simplex and Flow Control modes. They are implemented to control the transmission and reception between the Data Terminal Equipment (DTE). The UEN<1:0> bits in the UxMODE register configure these pins.

15.7 Infrared Support

The UARTx module provides two types of infrared UARTx support: one is the IrDA clock output to support an external IrDA encoder and decoder device (legacy module support), and the other is the full implementation of the IrDA encoder and decoder.

As the IrDA modes require a 16x baud clock, they will only work when the BRGH bit (UxMODE<3>) is '0'.

15.7.1 EXTERNAL IrDA SUPPORT – IrDA CLOCK OUTPUT

To support external IrDA encoder and decoder devices, the UxBCLK pin (same as the $\overline{\text{UxRTS}}$ pin) can be configured to generate the 16x baud clock. When UEN<1:0> = 11, the UxBCLK pin will output the 16x baud clock if the UARTx module is enabled; it can be used to support the IrDA codec chip.

15.7.2 BUILT-IN IrDA ENCODER AND DECODER

The UARTx has full implementation of the IrDA encoder and decoder as part of the UARTx module. The built-in IrDA encoder and decoder functionality is enabled using the IREN bit (UxMODE<12>). When enabled (IREN = 1), the receive pin (UxRX) acts as the input from the infrared receiver. The transmit pin (UxTX) acts as the output to the infrared transmitter.

FIGURE 16-2: 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						
0010 - Every 10 seconds						s
0011 - Every minute						s s
0100 - Every 10 minutes					m	s s
0101 - Every hour					m m	s s
0110 - Every day				h h	m m	s s
0111 - Every week	d			h h	m m	s s
1000 - Every month			d d	h h	m m	s s
1001 - Every year(1)		m m	d d	h h	m m	s s

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

16.5 Power Control

The RTCC includes a power control feature that allows the device to periodically wake-up an external device, wait for the device to be stable before sampling wake-up events from that device and then shut down the external device. This can be done completely autonomously by the RTCC, without the need to wake from the current low-power mode (Sleep, Deep Sleep, etc.).

To enable this feature, the RTCC must be enabled (RTCEN = 1), the PWCEN register bit must be set and the RTCC pin must be driving the PWC control signal (RTCOE = 1 and RTCCLK<1:0> = 11).

The polarity of the PWC control signal may be chosen using the PWCPOL register bit. Active-low or active-high may be used with the appropriate external switch to turn on or off the power to one or more external devices. The active-low setting may also be used in conjunction with an open-drain setting on the RTCC pin. This setting is able to drive the GND pin(s) of the external device directly (with the appropriate external V_{DD} pull-up device), without the need for external switches. Finally, the CHIME bit should be set to enable the PWC periodicity.

REGISTER 17-4: CLCxGLSL: CLCx GATE LOGIC INPUT SELECT LOW REGISTER (CONTINUED)

bit 3 **G1D2T:** Gate 1 Data Source 2 True Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 1
0 = The Data Source 2 inverted signal is disabled for Gate 1

bit 2 **G1D2N:** Gate 1 Data Source 2 Negated Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 1
0 = The Data Source 2 inverted signal is disabled for Gate 1

bit 1 **G1D1T:** Gate 1 Data Source 1 True Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 1
0 = The Data Source 1 inverted signal is disabled for Gate 1

bit 0 **G1D1N:** Gate 1 Data Source 1 Negated Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 1
0 = The Data Source 1 inverted signal is disabled for Gate 1

REGISTER 17-5: CLCxGLSH: CLCx GATE LOGIC INPUT SELECT HIGH REGISTER (CONTINUED)

bit 3 **G3D2T:** Gate 3 Data Source 2 True Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 3
0 = The Data Source 2 inverted signal is disabled for Gate 3

bit 2 **G3D2N:** Gate 3 Data Source 2 Negated Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 3
0 = The Data Source 2 inverted signal is disabled for Gate 3

bit 1 **G3D1T:** Gate 3 Data Source 1 True Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 3
0 = The Data Source 1 inverted signal is disabled for Gate 3

bit 0 **G3D1N:** Gate 3 Data Source 1 Negated Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 3
0 = The Data Source 1 inverted signal is disabled for Gate 3

19.0 12-BIT A/D CONVERTER WITH THRESHOLD DETECT

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 on the 12-Bit A/D Converter with Threshold Detect, refer to the *"PIC24F Family Reference Manual"*, **"12-Bit A/D Converter with Threshold Detect"** (DS39739).

The PIC24F 12-bit A/D Converter has the following key features:

- Successive Approximation Register (SAR) Conversion
- Conversion Speeds of up to 100 ksps
- Up to 32 Analog Input Channels (internal and external)
- Multiple Internal Reference Input Channels
- External Voltage Reference Input Pins
- Unipolar Differential Sample-and-Hold (S/H) Amplifier
- Automated Threshold Scan and Compare Operation to Pre-Evaluate Conversion Results
- Selectable Conversion Trigger Source
- Fixed-Length (one word per channel), Configurable Conversion Result Buffer
- Four Options for Results Alignment
- Configurable Interrupt Generation
- Operation During CPU Sleep and Idle modes

The 12-bit A/D Converter module is an enhanced version of the 10-bit module offered in some PIC24 devices. Both modules are Successive Approximation Register (SAR) converters at their cores, surrounded by a range of hardware features for flexible configuration. This version of the module extends functionality by providing 12-bit resolution, a wider range of automatic sampling options and tighter integration with other analog modules, such as the CTMU, and a configurable results buffer. There is a legacy 10-bit mode on this A/D to allow the option to run with lower resolution in order to obtain higher throughput. This module also includes a unique Threshold Detect feature that allows the module itself to make simple decisions based on the conversion results.

A simplified block diagram for the module is illustrated in Figure 19-1.

TABLE 19-4: NUMERICAL EQUIVALENTS OF VARIOUS RESULT CODES: 10-BIT FRACTIONAL FORMATS

VIN/VREF	10-Bit Differential Output Code (11-bit result)	16-Bit Fractional Format/ Equivalent Decimal Value		16-Bit Signed Fractional Format/ Equivalent Decimal Value	
+1023/1024	011 1111 1111	1111 1111 1100 0000	0.999	0111 1111 1110 0000	0.999
+1022/1024	011 1111 1110	1111 1111 1000 0000	0.998	0111 1111 1000 0000	0.998
•••					
+1/1024	000 0000 0001	0000 0000 0100 0000	0.001	0000 0000 0010 0000	0.001
0/1024	000 0000 0000	0000 0000 0000 0000	0.000	0000 0000 0000 0000	0.000
-1/1024	101 1111 1111	0000 0000 0000 0000	0.000	1111 1111 1110 0000	-0.001
•••					
-1023/1024	100 0000 0001	0000 0000 0000 0000	0.000	1000 0000 0010 0000	-0.999
-1024/1024	100 0000 0000	0000 0000 0000 0000	0.000	1000 0000 0000 0000	-1.000

The WDT, prescaler and postscaler are reset:

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

If the WDT is enabled in hardware (FWDTEN<1:0> = 11), it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake and code execution will continue from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE bit (RCON<3:2>) will need to be cleared in software after the device wakes up.

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

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

25.3.1 WINDOWED OPERATION

The Watchdog Timer has an optional Fixed Window mode of operation. In this Windowed mode, CLRWDT instructions can only reset the WDT during the last 1/4 of the programmed WDT period. A CLRWDT instruction executed before that window causes a WDT Reset, similar to a WDT time-out.

Windowed WDT mode is enabled by programming the Configuration bit, WINDIS (FWDT<6>), to '0'.

25.3.2 CONTROL REGISTER

The WDT is enabled or disabled by the FWDTEN<1:0> Configuration bits. When both of the FWDTEN<1:0> Configuration bits are set, the WDT is always enabled.

The WDT can be optionally controlled in software when the FWDTEN<1:0> Configuration bits have been programmed to '10'. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user to enable the WDT for critical code segments, and disable the WDT during non-critical segments, for maximum power savings. When the FWDTEN<1:0> bits are set to '01', the WDT is only enabled in Run and Idle modes, and is disabled in Sleep. Software control of the SWDTEN bit (RCON<5>) is disabled with this setting.

FIGURE 25-2: WDT BLOCK DIAGRAM

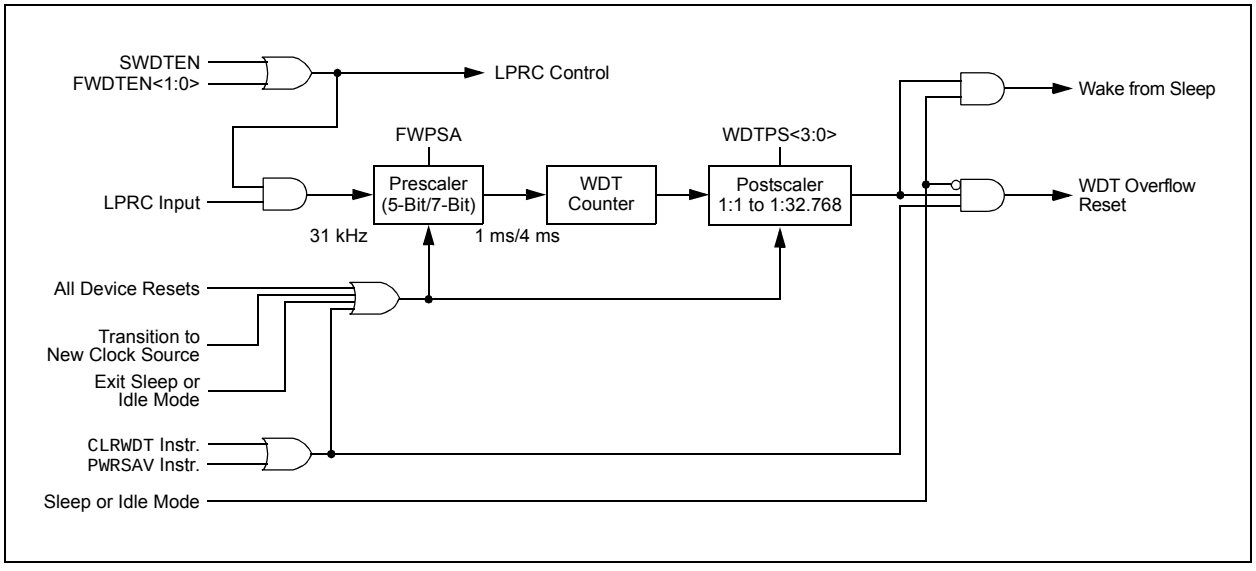
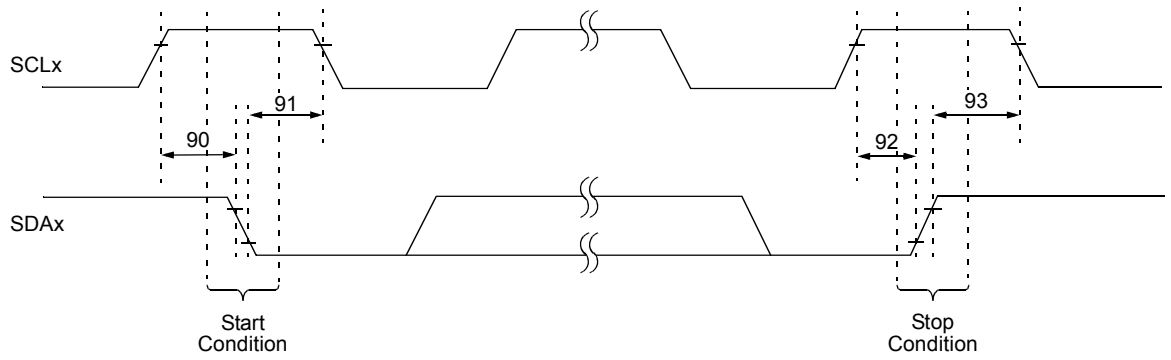


FIGURE 27-17: MSSPx I²C™ BUS START/STOP BITS TIMING WAVEFORMS



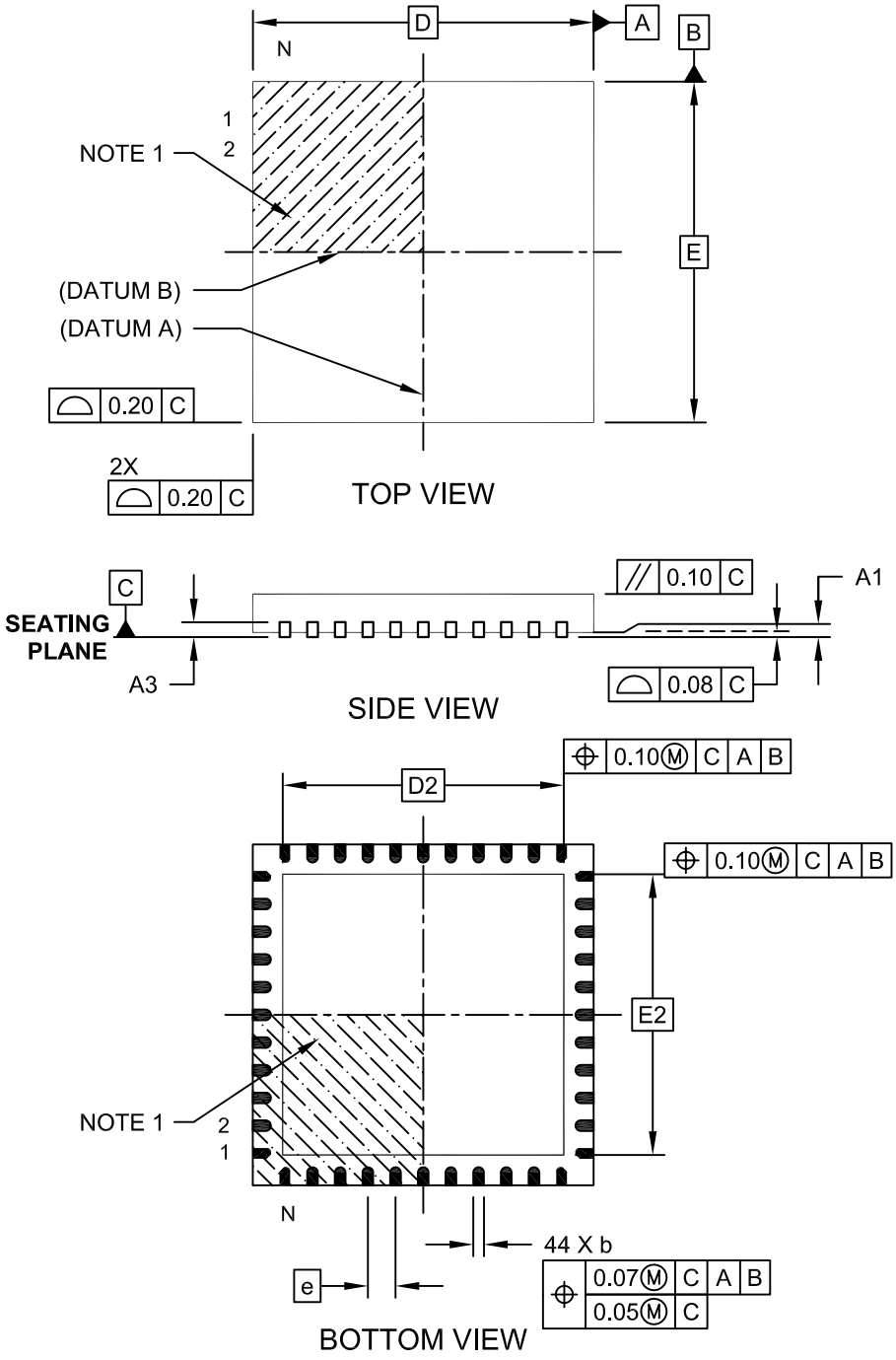
Note: Refer to Figure 27-5 for load conditions.

TABLE 27-35: I²C™ BUS START/STOP BITS REQUIREMENTS (MASTER MODE)

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
90	TSU:STA	Start Condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	2(TOSC)(BRG + 1)	—		
91	THD:STA	Start Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	2(TOSC)(BRG + 1)	—		
92	TSU:STO	Stop Condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		
93	THD:STO	Stop Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

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



Microchip Technology Drawing C04-103C Sheet 1 of 2