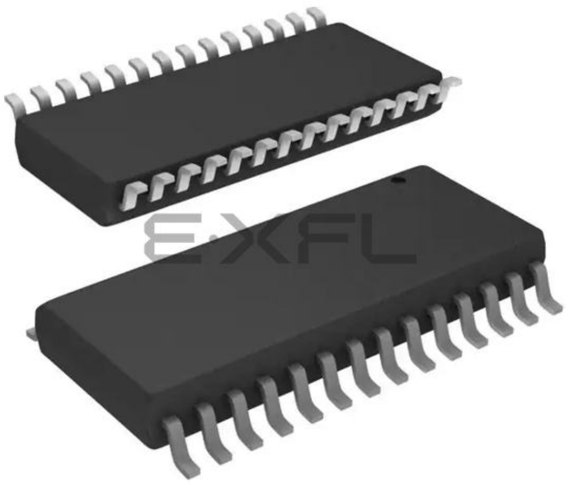


Welcome to E-XFL.COM

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	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18856-i-so

3.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- Program Memory
  - Configuration Words
  - Device ID
  - User ID
  - Program Flash Memory
- Data Memory
  - Core Registers
  - Special Function Registers
  - General Purpose RAM
  - Common RAM
  - Data EEPROM Memory

The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing
- NVMREG access

3.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing 32K x 14 program memory space. Table 3-1 shows the memory sizes implemented. Accessing a location above these boundaries will cause a wrap-around within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 3-1).

TABLE 3-1: DEVICE SIZES AND ADDRESSES

Device	Program Memory Size (Words)	Last Program Memory Address
PIC16(L)F18856/76	16,384	3FFFh

TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18856	PIC16(L)F18876	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
Bank 3													
						CPU CORE REGISTERS; see Table 3-2 for specifics							
18Ch	SSP1BUF			SSPBUF<7:0>								xxxx xxxx	xxxx xxxx
18Dh	SSP1ADD			SSPADD<7:0>								0000 0000	0000 0000
18Eh	SSP1MSK			SSPMSK<7:0>								1111 1111	1111 1111
18Fh	SSP1STAT			SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
190h	SSP1CON1			WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000
191h	SSP1CON2			GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
192h	SSP1CON3			ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
193h	—	—		Unimplemented								—	—
194h	—	—		Unimplemented								—	—
195h	—	—		Unimplemented								—	—
196h	SSP2BUF			SSPBUF<7:0>								xxxx xxxx	xxxx xxxx
197h	SSP2ADD			SSPADD<7:0>								0000 0000	0000 0000
198h	SSP2MSK			SSPMSK<7:0>								1111 1111	1111 1111
199h	SSP2STAT			SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
19Ah	SSP2CON1			WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000
19Bh	SSP2CON2			GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
19Ch	SSP2CON3			ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
19Dh	—	—		Unimplemented								—	—
19Eh	—	—		Unimplemented								—	—
19Fh	—	—		Unimplemented								—	—

Legend: x = unknown, u = unchanged, q =depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Register present on PIC16F18855/75 devices only.

2: Unimplemented, read as '1'.

4.6 Device ID and Revision ID

The 14-bit device ID word is located at 8006h and the 14-bit revision ID is located at 8005h. These locations are read-only and cannot be erased or modified.

Development tools, such as device programmers and debuggers, may be used to read the Device ID, Revision ID and Configuration Words. These locations can also be read from the NVMCON register.

4.7 Register Definitions: Device and Revision

REGISTER 4-6: DEVID: DEVICE ID REGISTER

R	R	R	R	R	R
DEV<13:8>					
bit 13					bit 8

R	R	R	R	R	R	R	R
DEV<7:0>							
bit 7							bit 0

Legend:	
R = Readable bit	
'1' = Bit is set	'0' = Bit is cleared

bit 13-0 **DEV<13:0>:** Device ID bits

Device	DEVID<13:0> Values
PIC16F18856	`11 0000 0111 0000 (3070h)`
PIC16LF18856	`11 0000 0111 0010 (3072h)`
PIC16F18876	`11 0000 0111 0001 (3071h)`
PIC16LF18876	`11 0000 0111 0011 (3073h)`

REGISTER 4-7: REVISIONID: REVISION ID REGISTER

R	R	R	R	R	R	R	R	R	R	R	R	R	R
1	0	MJRREV<5:0>						MNRREV<5:0>					
bit 13													bit 0

Legend:		
R = Readable bit		
'0' = Bit is cleared	'1' = Bit is set	x = Bit is unknown

bit 13-12 **Fixed Value**: Read-only bits
These bits are fixed with value '`10`' for all devices included in this data sheet.

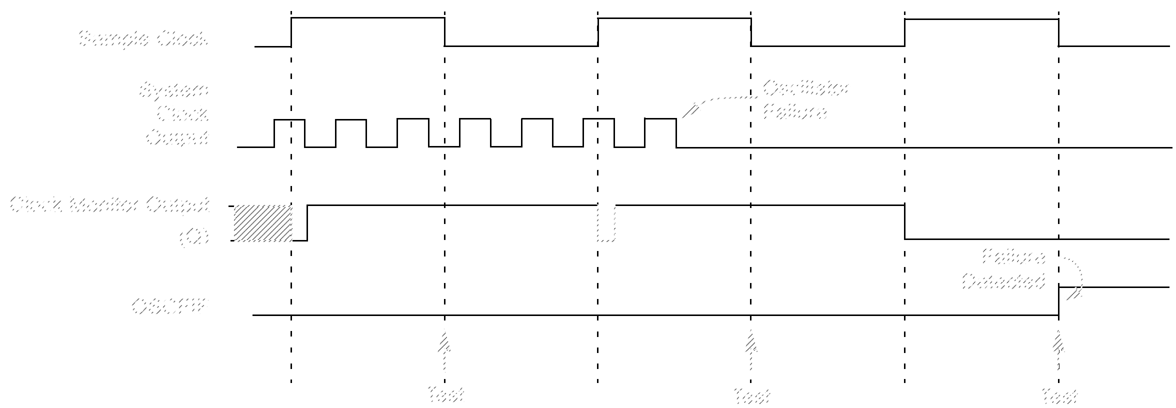
bit 11-6 **MJRREV<5:0>**: Major Revision ID bits
These bits are used to identify a major revision. A major revision is indicated by an all layer revision (B0, C0, etc.)

bit 5-0 **MNRREV<5:0>**: Minor Revision ID bits
These bits are used to identify a minor revision.

6.4.4 RESET OR WAKE-UP FROM SLEEP

The FSCM is designed to detect an oscillator failure after the Oscillator Start-up Timer (OST) has expired. The OST is used after waking up from Sleep and after any type of Reset. The OST is not used with the EC Clock modes so that the FSCM will be active as soon as the Reset or wake-up has completed. Therefore, the device will always be executing code while the OST is operating.

FIGURE 6-10: FSCM TIMING DIAGRAM



Note: The system clock is normally at a much higher frequency than the sample clock. The relative frequencies in this example have been chosen for clarity.

TABLE 7-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	134
PIE0	—	—	TMR0IE	IOCIE	—	—	—	INTE	135
PIE1	OSFIE	CSWIE	—	—	—	—	ADTIE	ADIE	136
PIE2	—	ZCDIE	—	—	—	—	C2IE	C1IE	137
PIE3	—	—	RCIE	TXIE	BCL2IE	SSP2IE	BCL1IE	SSP1IE	138
PIE4	—	—	TMR6IE	TMR5IE	TMR4IE	TMR3IE	TMR2IE	TMR1IE	139
PIE5	CLC4IE	CLC3IE	CLC2IE	CLC1IE	—	TMR5GIE	TMR3GIE	TMR1GIE	140
PIE6	—	—	—	CCP5IE	CCP4IE	CCP3IE	CCP2IE	CCP1IE	141
PIE7	SCANIE	CRCIE	NVMIE	NCO1IE	—	CWG3IE	CWG2IE	CWG1IE	142
PIE8	—	—	SMT2PWAIE	SMT2PRAIE	SMT2IE	SMT1PWAIE	SMT1PRAIE	SMT1IE	143
PIR0	—	—	TMR0IF	IOCIF	—	—	—	INTF	144
PIR1	OSFIF	CSWIF	—	—	—	—	ADTIF	ADIF	145
PIR2	—	ZCDIF	—	—	—	—	C2IF	C1IF	146
PIR3	—	—	RCIF	TXIF	BCL2IF	SSP2IF	BCL1IF	SSP1IF	147
PIR4	—	—	TMR6IF	TMR5IF	TMR4IF	TMR3IF	TMR2IF	TMR1IF	148
PIR5	CLC4IF	CLC3IF	CLC2IF	CLC1IF	—	TMR5GIF	TMR3GIF	TMR1GIF	149
PIR6	—	—	—	CCP5IF	CCP4IF	CCP3IF	CCP2IF	CCP1IF	150
PIR7	SCANIF	CRCIF	NVMIF	NCO1IF	—	CWG3IF	CWG2IF	CWG1IF	152
PIR8	—	—	SMT2PWAIF	SMT2PRAIF	SMT2IF	SMT1PWAIF	SMT1PRAIF	SMT1IF	153

Legend: — = unimplemented location, read as ‘0’. Shaded cells are not used by interrupts.

8.1.2 INTERRUPTS DURING DOZE

If an interrupt occurs and the Recover-On-Interrupt bit is clear (ROI = 0) at the time of the interrupt, the Interrupt Service Routine (ISR) continues to execute at the rate selected by DOZE<2:0>. Interrupt latency is extended by the DOZE<2:0> ratio.

If an interrupt occurs and the ROI bit is set (ROI = 1) at the time of the interrupt, the DOZEN bit is cleared and the CPU executes at full speed. The prefetched instruction is executed and then the interrupt vector sequence is executed. In Figure 8-1, the interrupt occurs during the 2nd instruction cycle of the Doze period, and immediately brings the CPU out of Doze. If the Doze-On-Exit (DOE) bit is set (DOE = 1) when the RETFIE operation is executed, DOZEN is set, and the CPU executes at the reduced rate based on the DOZE<2:0> ratio.

8.2 Sleep Mode

Sleep mode is entered by executing the SLEEP instruction, while the Idle Enable (IDLEN) bit of the CPUDOZE register is clear (IDLEN = 0). If the SLEEP instruction is executed while the IDLEN bit is set (IDLEN = 1), the CPU will enter the IDLE mode (**Section 8.2.3 "Low-Power Sleep Mode"**).

Upon entering Sleep mode, the following conditions exist:

1. WDT will be cleared but keeps running if enabled for operation during Sleep
2. The $\overline{PD}$ bit of the STATUS register is cleared
3. The $\overline{TO}$ bit of the STATUS register is set
4. The CPU clock is disabled
5. 31 kHz LFINTOSC, HFINTOSC and SOSC are unaffected and peripherals using them may continue operation in Sleep.
6. Timer1 and peripherals that use it continue to operate in Sleep when the Timer1 clock source selected is:
   - LFINTOSC
   - T1CKI
   - Secondary Oscillator
7. ADC is unaffected if the dedicated FRC oscillator is selected
8. I/O ports maintain the status they had before Sleep was executed (driving high, low, or high-impedance)
9. Resets other than WDT are not affected by Sleep mode

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using any oscillator

I/O pins that are high-impedance inputs should be pulled to VDD or VSS externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules. See **Section 25.0 "5-Bit Digital-to-Analog Converter (DAC1) Module"** and **16.0 "Fixed Voltage Reference (FVR)"** for more information on these modules.

8.2.1 WAKE-UP FROM SLEEP

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{MCLR}$ pin, if enabled.
2. BOR Reset, if enabled.
3. POR Reset.
4. Watchdog Timer, if enabled.
5. Any external interrupt.
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information).

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 5.11 "Determining the Cause of a Reset"**.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes-up from Sleep, regardless of the source of wake-up.

REGISTER 18-2: CMxCON1: COMPARATOR Cx CONTROL REGISTER 1

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
—	—	—	—	—	—	INTP	INTN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-2 **Unimplemented:** Read as ‘0’

bit 1 **INTP:** Comparator Interrupt on Positive-Going Edge Enable bits
1 = The CxIF interrupt flag will be set upon a positive-going edge of the CxOUT bit
0 = No interrupt flag will be set on a positive-going edge of the CxOUT bit

bit 0 **INTN:** Comparator Interrupt on Negative-Going Edge Enable bits
1 = The CxIF interrupt flag will be set upon a negative-going edge of the CxOUT bit
0 = No interrupt flag will be set on a negative-going edge of the CxOUT bit

23.5.5 BURST AVERAGE MODE

The Burst Average mode (ADMD = '011') acts the same as the Average mode in most respects. The one way it differs is that it continuously retriggers ADC sampling until the ADCNT value is greater than or equal to ADRPT, even if Continuous Sampling mode (see **Section 23.5.8 "Continuous Sampling Mode"**) is not enabled. This allows for a threshold comparison on the average of a short burst of ADC samples.

23.5.6 LOWPASS FILTER MODE

The Lowpass Filter mode (ADMD = '100') acts similarly to the Average mode in how it handles samples (accumulates samples until ADCNT value greater than or equal to ADRPT, then triggers threshold comparison), but instead of a simple average, it performs a lowpass filter operation on all of the samples, reducing the effect of high-frequency noise on the average, then performs a threshold comparison on the results. (see Table 23-3 for a more detailed description of the mathematical operation). In this mode, the ADCRS bits determine the cut-off frequency of the lowpass filter (as demonstrated by Table 23-4).

23.5.7 THRESHOLD COMPARISON

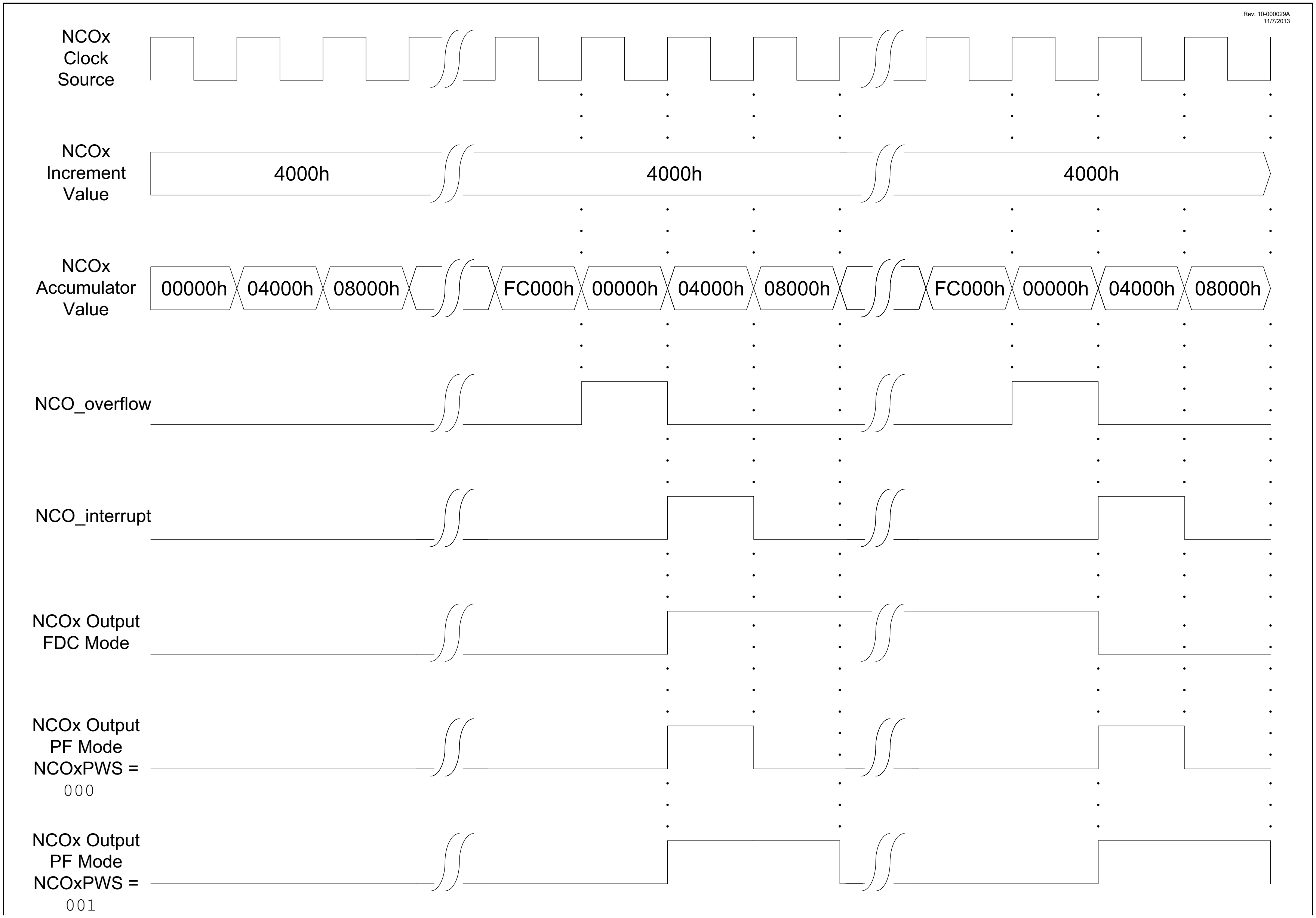
At the end of each computation:

- The conversion results are latched and held stable at the end-of-conversion.
- The difference value is calculated based on a difference calculation which is selected by the ADCALC<2:0> bits in the ADCON3 register. The value can be one of the following calculations (see Register 23-4 for more details):
  - The first derivative of single measurements
  - The CVD result in CVD mode
  - The current result vs. a setpoint
  - The current result vs. the filtered/average result
  - The first derivative of the filtered/average value
  - Filtered/average value vs. a setpoint
- The result of the calculation (ADERR) is compared to the upper and lower thresholds, ADUTH<ADUTHH:ADUTHL> and ADLTH<ADLTHH:ADLTHL> registers, to set the ADUTHR and ADLTHR flag bits. The threshold logic is selected by ADTMD<2:0> bits in the ADCON3 register. The threshold trigger option can be one of the following
  - Never interrupt
  - Error is less than lower threshold
  - Error is greater than or equal to lower threshold
  - Error is between thresholds (inclusive)
  - Error is outside of thresholds
  - Error is less than or equal to upper threshold
  - Error is greater than upper threshold
  - Always interrupt regardless of threshold test results
- The threshold interrupt flag ADTIF is set when the threshold condition is met.

Note 1: The threshold tests are signed operations.

2: If ADAOV is set, a threshold interrupt is signaled.

FIGURE 24-2: FDC OUTPUT MODE OPERATION DIAGRAM



REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **T0EN:** TMR0 Enable bit
1 = The module is enabled and operating
0 = The module is disabled and in the lowest power mode

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

bit 5 **T0OUT:** TMR0 Output bit (read-only)
TMR0 output bit

bit 4 **T016BIT:** TMR0 Operating as 16-bit Timer Select bit
1 = TMR0 is a 16-bit timer
0 = TMR0 is an 8-bit timer

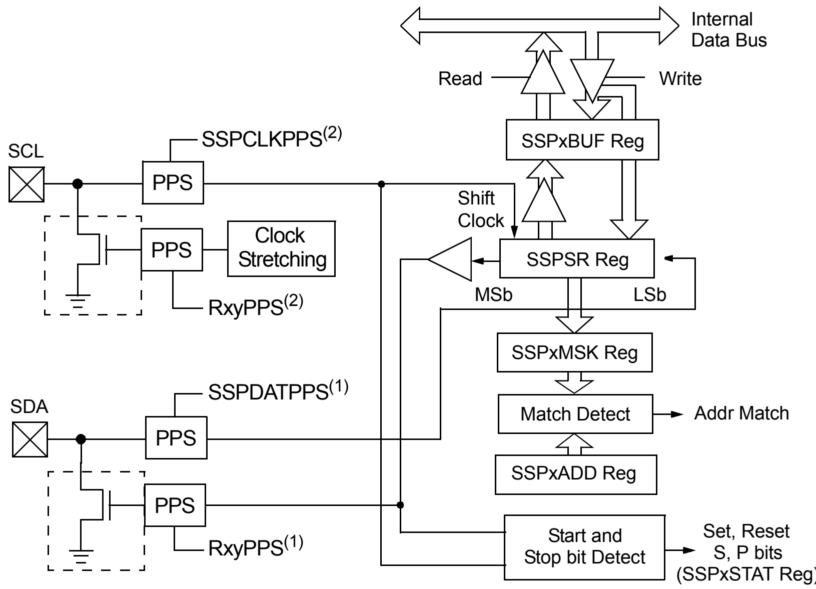
bit 3-0 **T0OUTPS<3:0>:** TMR0 output postscaler (divider) select bits
1111 = 1:16 Postscaler
1110 = 1:15 Postscaler
1101 = 1:14 Postscaler
1100 = 1:13 Postscaler
1011 = 1:12 Postscaler
1010 = 1:11 Postscaler
1001 = 1:10 Postscaler
1000 = 1:9 Postscaler
0111 = 1:8 Postscaler
0110 = 1:7 Postscaler
0101 = 1:6 Postscaler
0100 = 1:5 Postscaler
0011 = 1:4 Postscaler
0010 = 1:3 Postscaler
0001 = 1:2 Postscaler
0000 = 1:1 Postscaler

TABLE 29-3: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CCP1CON	EN	—	OUT	FMT	MODE<3:0>				452
CCP2CON	EN	—	OUT	FMT	MODE<3:0>				452
CCPTMRS0	C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>		455
CCPTMRS1	—	—	P7TSEL<1:0>		P6TSEL<1:0>		C5TSEL<1:0>		455
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	134
PIE1	OSFIE	CSWIE	—	—	—	—	ADTIE	ADIE	136
PIR1	OSFIF	CSWIF	—	—	—	—	ADTIF	ADIF	145
PR2	Timer2 Module Period Register								425*
TMR2	Holding Register for the 8-bit TMR2 Register								425*
T2CON	ON	CKPS<2:0>			OUTPS<3:0>				441
T2CLKCON	—	—	—	—	CS<3:0>				440
T2RST	—	—	—	RSEL<4:0>					443
T2HLT	PSYNC	CKPOL	CKSYNC	—	MODE<3:0>				442
PR4	Timer4 Module Period Register								425*
TMR4	Holding Register for the 8-bit TMR4 Register								425*
T4CON	ON	CKPS<2:0>			OUTPS<3:0>				441
T4CLKCON	—	—	—	—	—	CS<3:0>			440
T4RST	—	—	—	RSEL<4:0>					443
T4HLT	PSYNC	CKPOL	CKSYNC	—	MODE<3:0>				442
PR6	Timer6 Module Period Register								425*
TMR6	Holding Register for the 8-bit TMR6 Register								425*
T6CON	ON	CKPS<2:0>			OUTPS<3:0>				441
T6CLKCON	—	—	—	—	—	CS<2:0>			440
T6RST	—	—	—	RSEL<4:0>					443
T6HLT	PSYNC	CKPOL	CKSYNC	—	MODE<3:0>				442

Legend: — = unimplemented location, read as ‘0’. Shaded cells are not used for Timer2 module.
* Page provides register information.

FIGURE 31-3: MSSP BLOCK DIAGRAM (I²C SLAVE MODE)



Note 1: SDA pin selections must be the same for input and output

2: SCL pin selections must be the same for input and output

REGISTER 32-10: SMTxCPRL: SMT CAPTURED PERIOD REGISTER – LOW BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<7:0>**: Significant bits of the SMT Period Latch – Low Byte

REGISTER 32-11: SMTxCPRH: SMT CAPTURED PERIOD REGISTER – HIGH BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<15:8>**: Significant bits of the SMT Period Latch – High Byte

REGISTER 32-12: SMTxCPRU: SMT CAPTURED PERIOD REGISTER – UPPER BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<23:16>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<23:16>**: Significant bits of the SMT Period Latch – Upper Byte

33.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR3 register is set whenever the EUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE, Interrupt Enable bit of the PIE3 register
- PEIE, Peripheral Interrupt Enable bit of the INTCON register
- GIE, Global Interrupt Enable bit of the INTCON register

The RCIF interrupt flag bit will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

33.1.2.4 Receive Framing Error

Each character in the receive FIFO buffer has a corresponding framing error Status bit. A framing error indicates that a Stop bit was not seen at the expected time. The framing error status is accessed via the FERR bit of the RC1STA register. The FERR bit represents the status of the top unread character in the receive FIFO. Therefore, the FERR bit must be read before reading the RCREG.

The FERR bit is read-only and only applies to the top unread character in the receive FIFO. A framing error (FERR = 1) does not preclude reception of additional characters. It is not necessary to clear the FERR bit. Reading the next character from the FIFO buffer will advance the FIFO to the next character and the next corresponding framing error.

The FERR bit can be forced clear by clearing the SPEN bit of the RC1STA register which resets the EUSART. Clearing the CREN bit of the RC1STA register does not affect the FERR bit. A framing error by itself does not generate an interrupt.

Note:	If all receive characters in the receive FIFO have framing errors, repeated reads of the RCREG will not clear the FERR bit.

33.1.2.5 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before the FIFO is accessed. When this happens the OERR bit of the RC1STA register is set. The characters already in the FIFO buffer can be read but no additional characters will be received until the error is cleared. The error must be cleared by either clearing the CREN bit of the RC1STA register or by resetting the EUSART by clearing the SPEN bit of the RC1STA register.

33.1.2.6 Receiving 9-Bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RC1STA register is set the EUSART will shift nine bits into the RSR for each character received. The RX9D bit of the RC1STA register is the ninth and Most Significant data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

33.1.2.7 Address Detection

A special Address Detection mode is available for use when multiple receivers share the same transmission line, such as in RS-485 systems. Address detection is enabled by setting the ADDEN bit of the RC1STA register.

Address detection requires 9-bit character reception. When address detection is enabled, only characters with the ninth data bit set will be transferred to the receive FIFO buffer, thereby setting the RCIF interrupt bit. All other characters will be ignored.

Upon receiving an address character, user software determines if the address matches its own. Upon address match, user software must disable address detection by clearing the ADDEN bit before the next Stop bit occurs. When user software detects the end of the message, determined by the message protocol used, software places the receiver back into the Address Detection mode by setting the ADDEN bit.

33.1.2.8 Asynchronous Reception Setup:

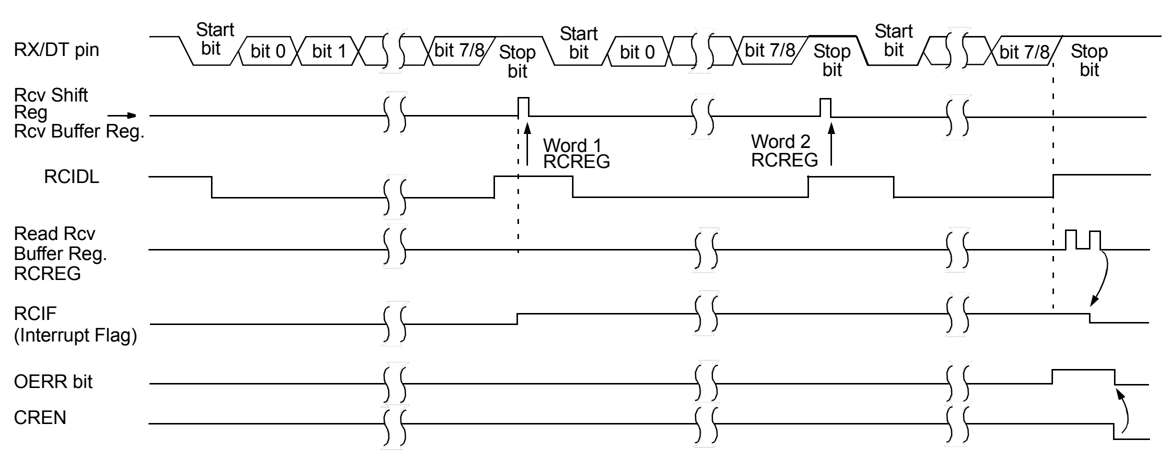
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 33.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RC1STA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

33.1.2.9 9-bit Address Detection Mode Setup

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 33.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RC1STA register to get the error flags. The ninth data bit will always be set.
10. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 33-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

MOVWI — Move W to INDFn

Syntax: [*label*] MOVWI ++FSRn
[*label*] MOVWI --FSRn
[*label*] MOVWI FSRn++
[*label*] MOVWI FSRn--
[*label*] MOVWI k[FSRn]

Operands: n ∈ [0,1]
mm ∈ [00,01, 10, 11]
-32 ≤ k ≤ 31

Operation: W → INDFn
Effective address is determined by
- FSR + 1 (preincrement)
- FSR - 1 (predecrement)
- FSR + k (relative offset)

After the Move, the FSR value will be either:
- FSR + 1 (all increments)
- FSR - 1 (all decrements)

Unchanged

Status Affected: None

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	--FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn--	11

Description: This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

Note: The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h-FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.

The increment/decrement operation on FSRn WILL NOT affect any Status bits.

NOP — No Operation

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

Words: 1

Cycles: 1

Example:
```
NOP
```

RESET — Software Reset

Syntax: [*label*] RESET

Operands: None

Operation: Execute a device Reset. Resets the $\overline{\text{RI}}$ flag of the PCON register.

Status Affected: None

Description: This instruction provides a way to execute a hardware Reset by software.

RETFIE — Return from Interrupt

Syntax: [*label*] RETFIE k

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

Description: Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a 2-cycle instruction.

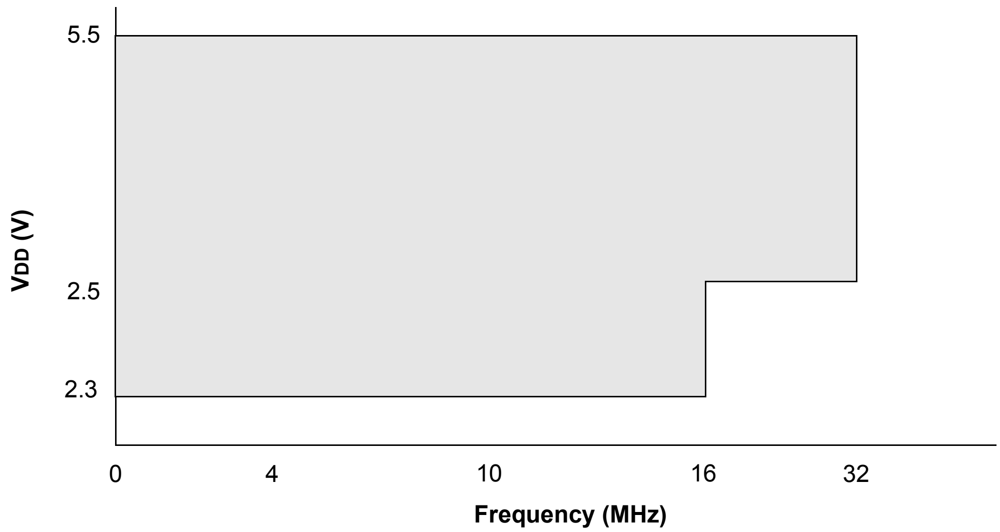
Words: 1

Cycles: 2

Example:
```
RETFIE
```

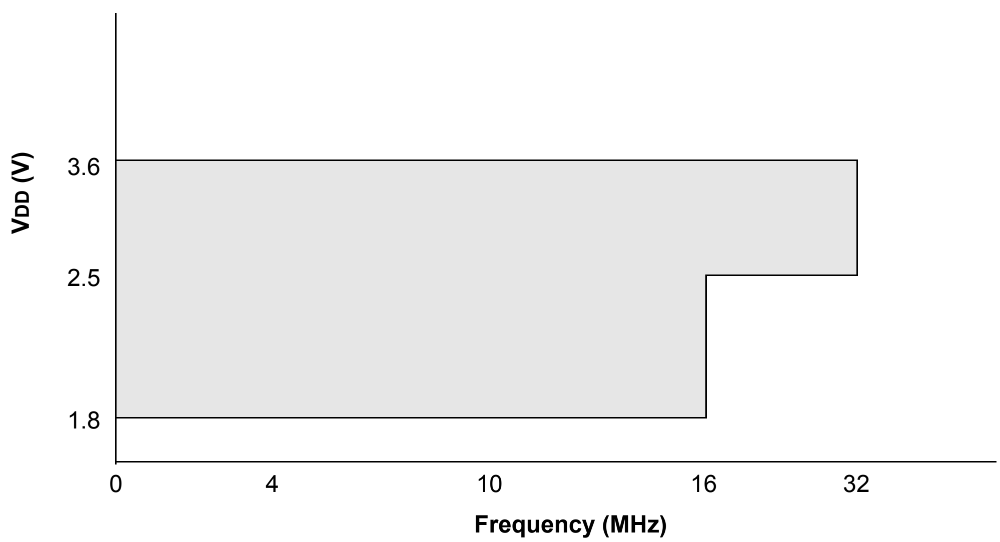
After Interrupt
PC = TOS
GIE = 1

FIGURE 37-1: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16F18856/76 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
2: Refer to Table for each Oscillator mode's supported frequencies.

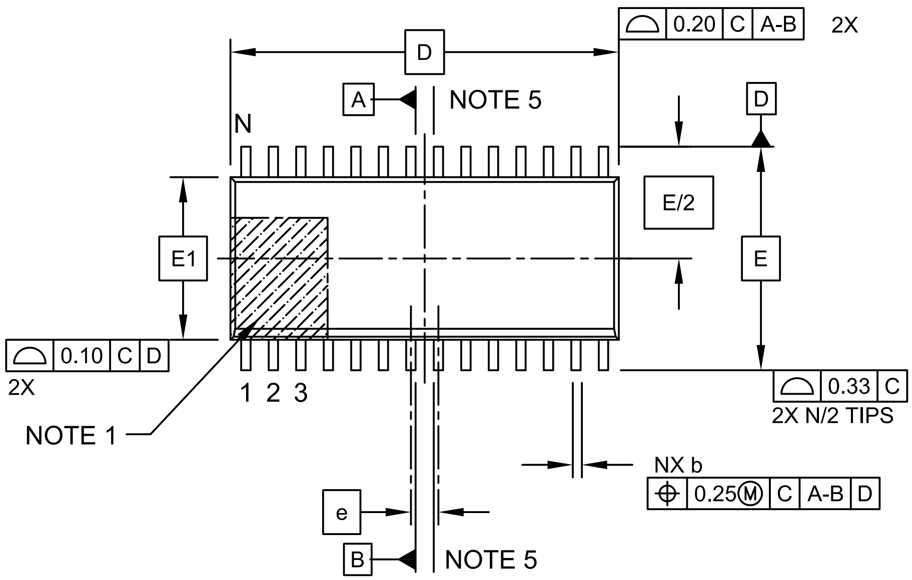
FIGURE 37-2: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16LF18856/76 ONLY



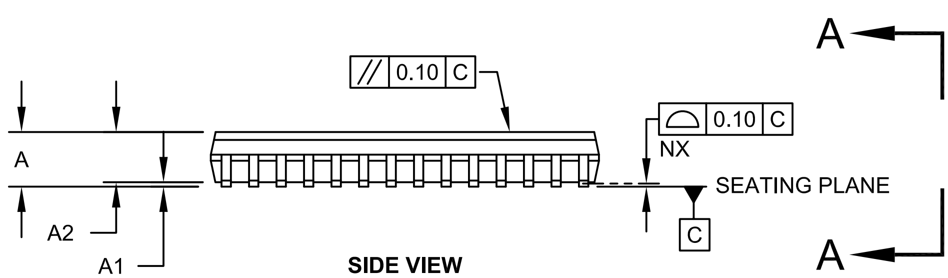
Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
2: Refer to Table for each Oscillator mode's supported frequencies.

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

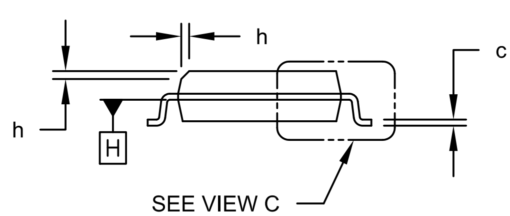
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

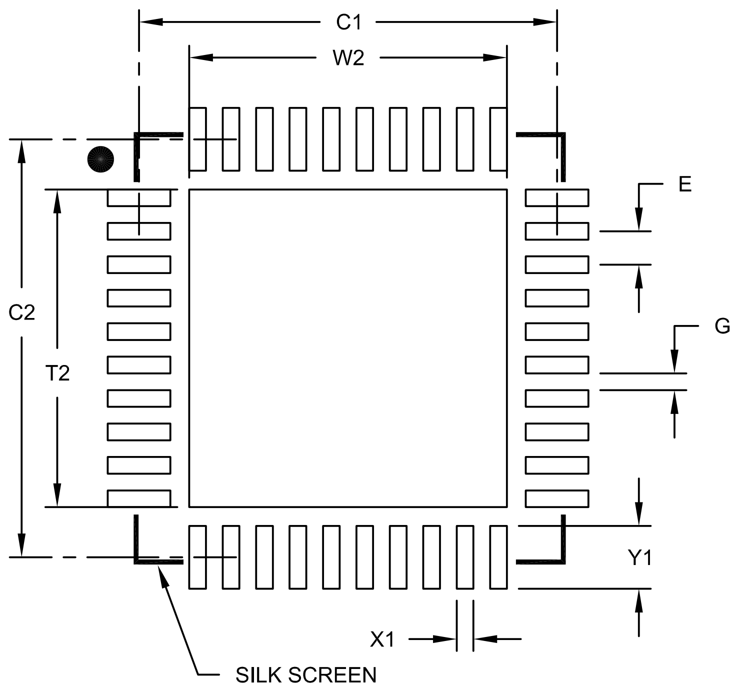


VIEW A-A

Microchip Technology Drawing C04-052C Sheet 1 of 2

40-Lead Plastic Ultra Thin Quad Flat, No Lead Package (MV) - 5x5 mm Body [UQFN]

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		0.40 BSC	
Optional Center Pad Width	W2			3.80
Optional Center Pad Length	T2			3.80
Contact Pad Spacing	C1		5.00	
Contact Pad Spacing	C2		5.00	
Contact Pad Width (X40)	X1			0.20
Contact Pad Length (X40)	Y1			0.75
Distance Between Pads	G	0.20		

Notes:

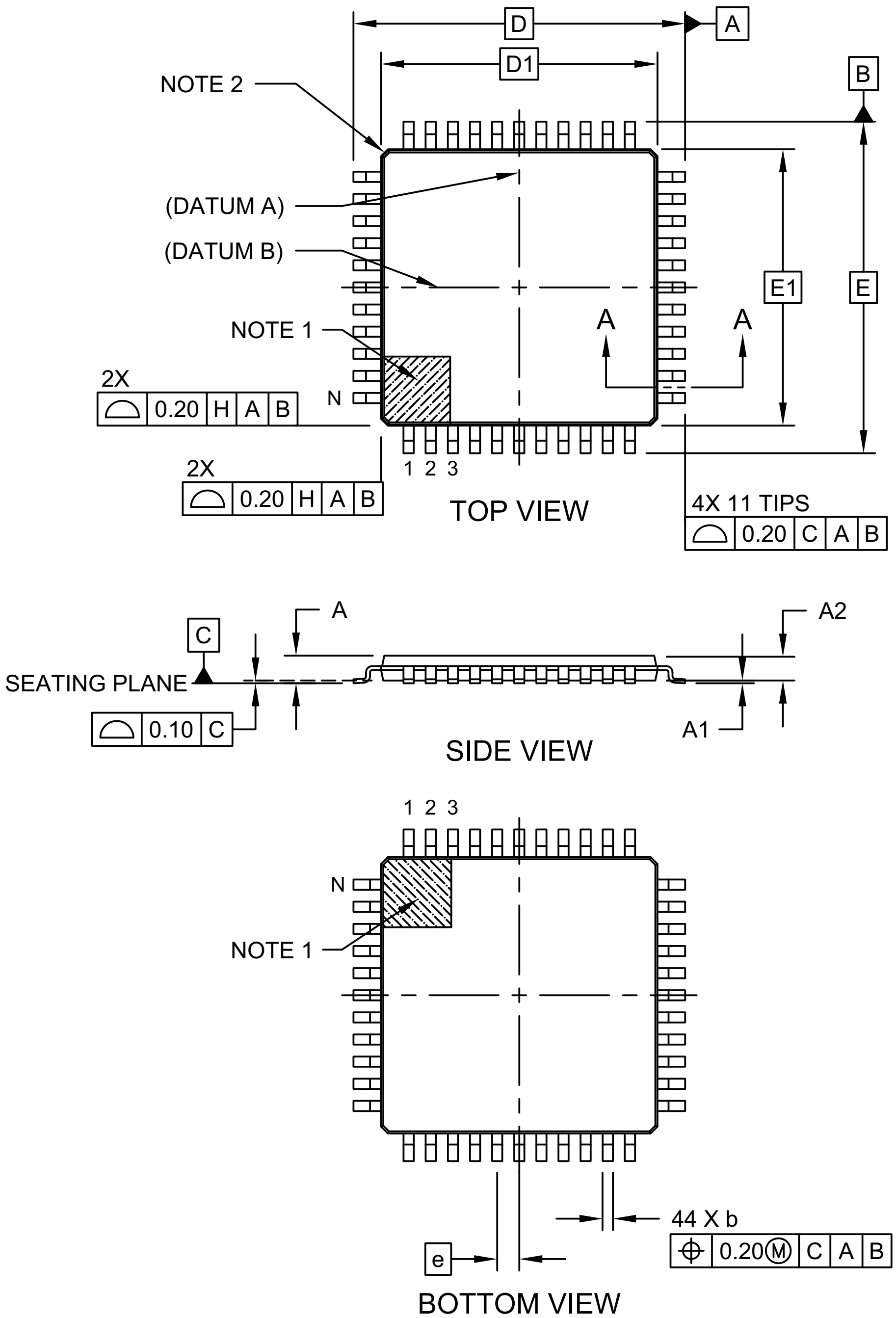
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2156B

44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [TQFP]

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



Microchip Technology Drawing C04-076C Sheet 1 of 2