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	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	18
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f687-e-p

1.0 DEVICE OVERVIEW

The PIC16F631/677/685/687/689/690 devices are covered by this data sheet. They are available in 20-pin PDIP, SOIC, TSSOP and QFN packages.

Block Diagrams and pinout descriptions of the devices are as follows:

- PIC16F631 (Figure 1-1, Table 1-1)
- PIC16F677 (Figure 1-2, Table 1-2)
- PIC16F685 (Figure 1-3, Table 1-3)
- PIC16F687/PIC16F689 (Figure 1-4, Table 1-4)
- PIC16F690 (Figure 1-5, Table 1-5)

FIGURE 1-1: PIC16F631 BLOCK DIAGRAM

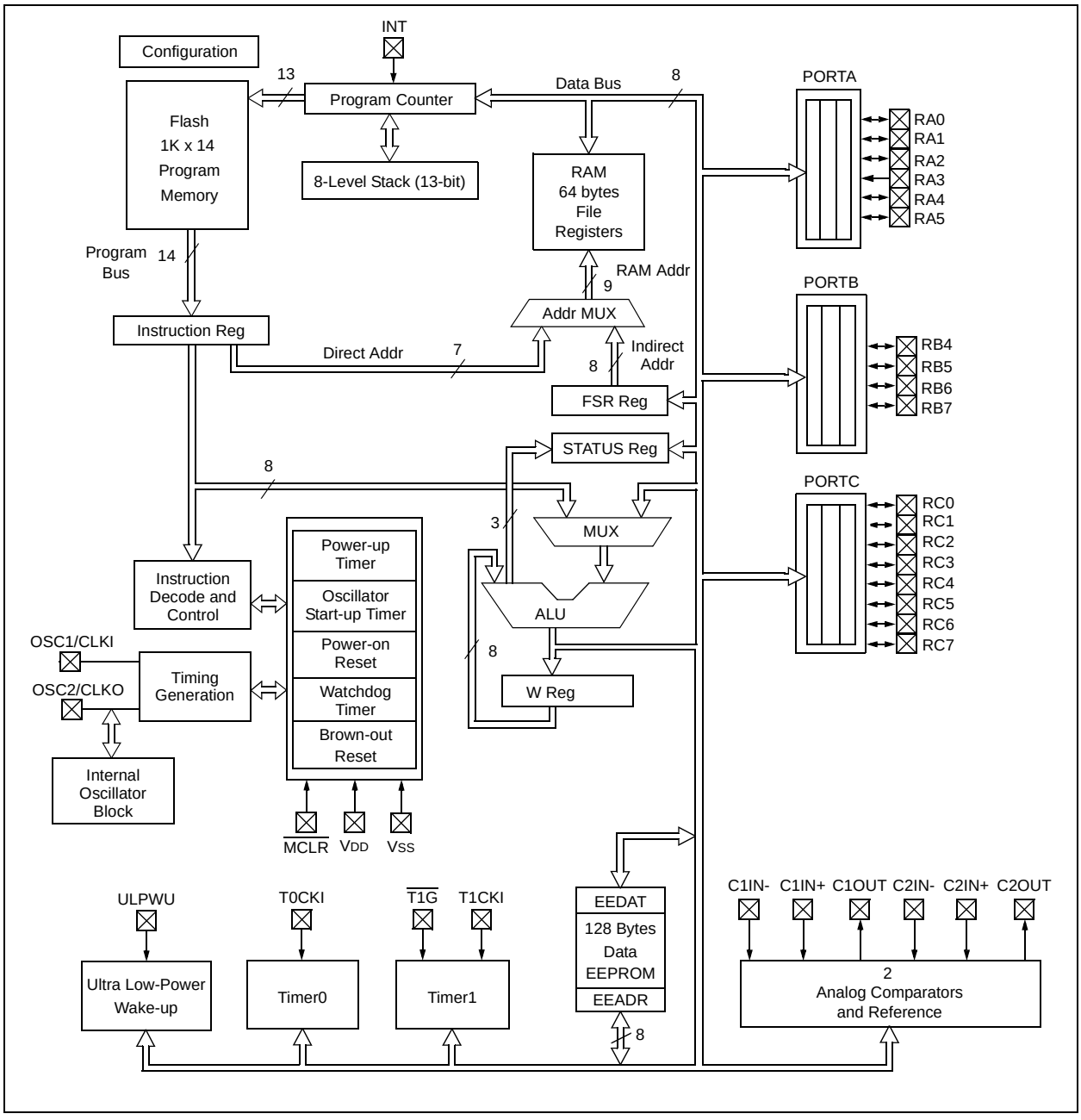


TABLE 1-1: PINOUT DESCRIPTION – PIC16F631

Name	Function	Input Type	Output Type	Description
RA0/C1IN+/ICSPDAT/ULPWU	RA0	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	C1IN+	AN	—	Comparator C1 non-inverting input.
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.
	ULPWU	AN	—	Ultra Low-Power Wake-up input.
RA1/C12IN0-/ICSPCLK	RA1	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	C12IN0-	AN	—	Comparator C1 or C2 inverting input.
	ICSPCLK	ST	—	ICSP™ clock.
RA2/T0CKI/INT/C1OUT	RA2	ST	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	T0CKI	ST	—	Timer0 clock input.
	INT	ST	—	External interrupt pin.
	C1OUT	—	CMOS	Comparator C1 output.
RA3/$\overline{\text{MCLR}}$/V_{PP}	RA3	TTL	—	General purpose input. Individually controlled interrupt-on-change.
	$\overline{\text{MCLR}}$	ST	—	Master Clear with internal pull-up.
	V_{PP}	HV	—	Programming voltage.
RA4/$\overline{\text{T1G}}$/OSC2/CLKOUT	RA4	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	$\overline{\text{T1G}}$	ST	—	Timer1 gate input.
	OSC2	—	XTAL	Crystal/Resonator.
	CLKOUT	—	CMOS	F_{OSC}/4 output.
RA5/T1CKI/OSC1/CLKIN	RA5	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	T1CKI	ST	—	Timer1 clock input.
	OSC1	XTAL	—	Crystal/Resonator.
	CLKIN	ST	—	External clock input/RC oscillator connection.
RB4	RB4	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
RB5	RB5	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
RB6	RB6	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
RB7	RB7	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
RC0/C2IN+	RC0	ST	CMOS	General purpose I/O.
	C2IN+	AN	—	Comparator C2 non-inverting input.
RC1/C12IN1-	RC1	ST	CMOS	General purpose I/O.
	C12IN1-	AN	—	Comparator C1 or C2 inverting input.
RC2/C12IN2-	RC2	ST	CMOS	General purpose I/O.
	C12IN2-	AN	—	Comparator C1 or C2 inverting input.
RC3/C12IN3-	RC3	ST	CMOS	General purpose I/O.
	C12IN3-	AN	—	Comparator C1 or C2 inverting input.
RC4/C2OUT	RC4	ST	CMOS	General purpose I/O.
	C2OUT	—	CMOS	Comparator C2 output.
RC5	RC5	ST	CMOS	General purpose I/O.

Legend: AN = Analog input or output; CMOS=CMOS compatible input or output
TTL = TTL compatible input; ST= Schmitt Trigger input with CMOS levels
HV = High Voltage; XTAL= Crystal

TABLE 2-1: PIC16F631/677/685/687/689/690 SPECIAL FUNCTION REGISTERS SUMMARY BANK 0

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 0											
00h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	43,200
01h	TMR0	Timer0 Module Register								xxxx xxxx	79,200
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	43,200
03h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	35,200
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	43,200
05h	PORTA(7)	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	57,200
06h	PORTB(7)	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	67,200
07h	PORTC(7)	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	74,200
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of Program Counter					---0 0000	43,200
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RABIE	T0IF	INTF	RABIF(1)	0000 000x	37,200
0Ch	PIR1	—	ADIF(4)	RCIF(2)	TXIF(2)	SSPIF(5)	CCP1IF(3)	TMR2IF(3)	TMR1IF	-000 0000	40,200
0Dh	PIR2	OSFIF	C2IF	C1IF	EEIF	—	—	—	—	0000 ----	41,200
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	85,200
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	85,200
10h	T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	0000 0000	87,200
11h	TMR2(3)	Timer2 Module Register								0000 0000	89,200
12h	T2CON(3)	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	90,200
13h	SSPBUF(5)	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	178,200
14h	SSPCON(5, 6)	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	177,200
15h	CCPR1L(3)	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	126,200
16h	CCPR1H(3)	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	126,200
17h	CCP1CON(3)	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	125,200
18h	RCSTA(2)	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	158,200
19h	TXREG(2)	EUSART Transmit Data Register								0000 0000	150
1Ah	RCREG(2)	EUSART Receive Data Register								0000 0000	155
1Bh	—	Unimplemented								—	—
1Ch	PWM1CON(3)	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	143,200
1Dh	ECCPAS(3)	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000	140,200
1Eh	ADRESH(4)	A/D Result Register High Byte								xxxx xxxx	113,200
1Fh	ADCON0(4)	ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	0000 0000	111,200

Legend: – = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: $\overline{MCLR}$ and WDT Reset do not affect the previous value data latch. The RABIF bit will be cleared upon Reset but will set again if the mismatch exists.

2: PIC16F687/PIC16F689/PIC16F690 only.

3: PIC16F685/PIC16F690 only.

4: PIC16F677/PIC16F685/PIC16F687/PIC16F689/PIC16F690 only.

5: PIC16F677/PIC16F687/PIC16F689/PIC16F690 only.

6: When SSPCON register bits SSPM<3:0> = 1001, any reads or writes to the SSPADD SFR address are accessed through the SSPMSK register. See Registers 13-2 and 13-3 for more detail.

7: Port pins with analog functions controlled by the ANSEL and ANSELH registers will read '0' immediately after a Reset even though the data latches are either undefined (POR) or unchanged (other Resets).

FIGURE 3-9: FSCM TIMING DIAGRAM

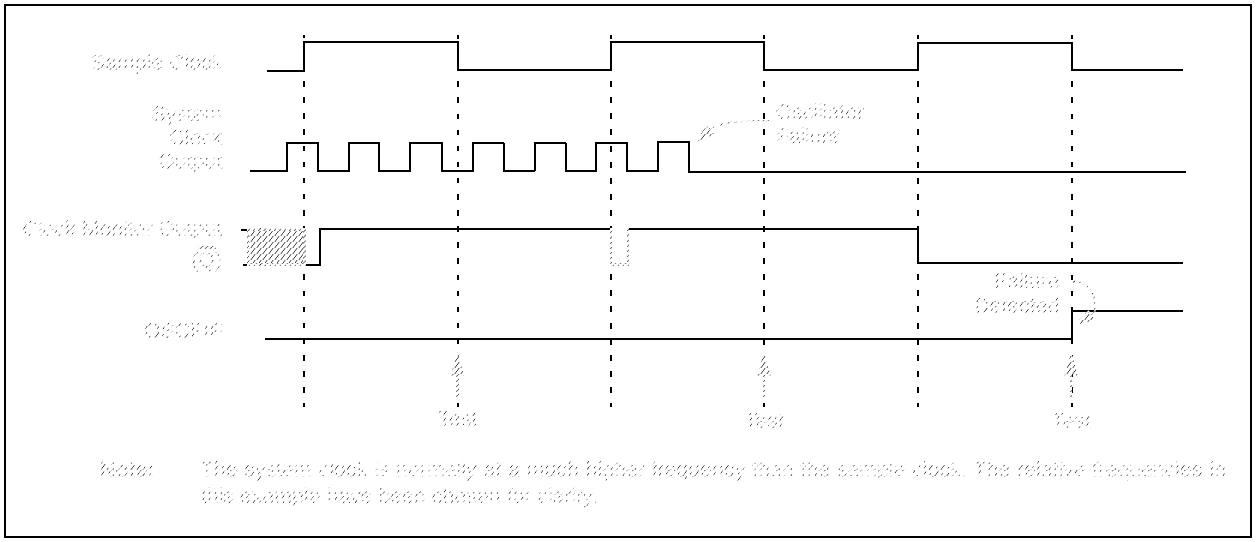


TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

Name	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(1)
CONFIG(2)	CPD	CP	MCLRE	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0	—	—
OSCCON	—	IRCF2	IRCF1	IRCF0	OSTS	HTS	LTS	SCS	-110 x000	-110 x000
OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	---u uuuu
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000

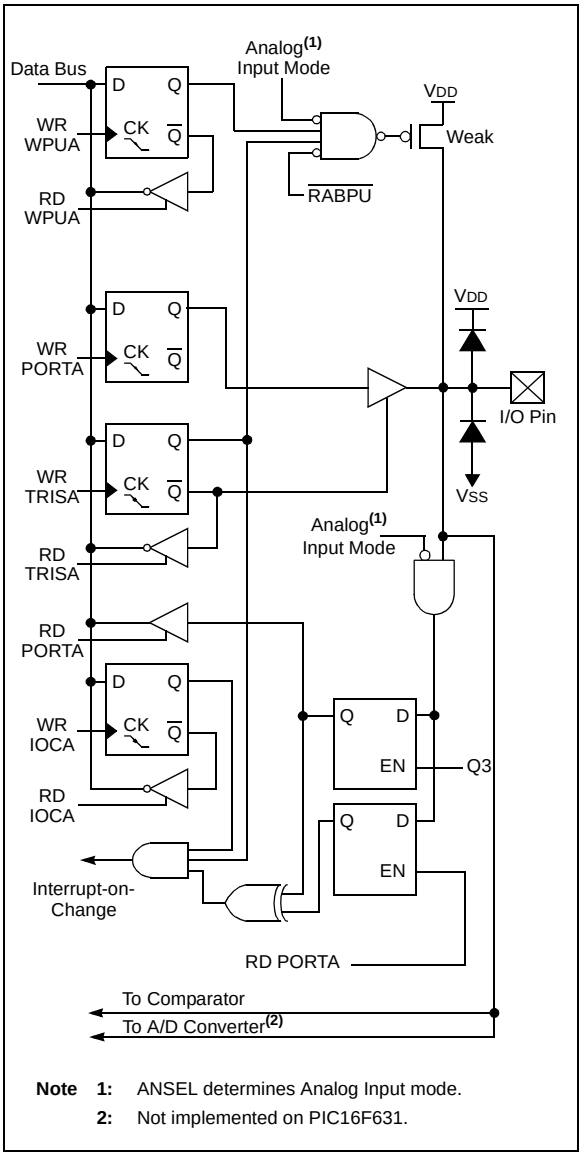
Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by oscillators.
Note 1: Other (non Power-up) Resets include $\overline{\text{MCLR}}$ Reset and Watchdog Timer Reset during normal operation.
2: See Configuration Word register (Register 14-1) for operation of all register bits.

4.2.5.2 RA1/AN1/C12IN0-/VREF/ICSPCLK

Figure 4-2 shows the diagram for this pin. The RA1/AN1/C12IN0-/VREF/ICSPCLK pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC (except PIC16F631)
- an analog input to Comparator C1 or C2
- a voltage reference input for the ADC
- In-Circuit Serial Programming clock

FIGURE 4-2: BLOCK DIAGRAM OF RA1



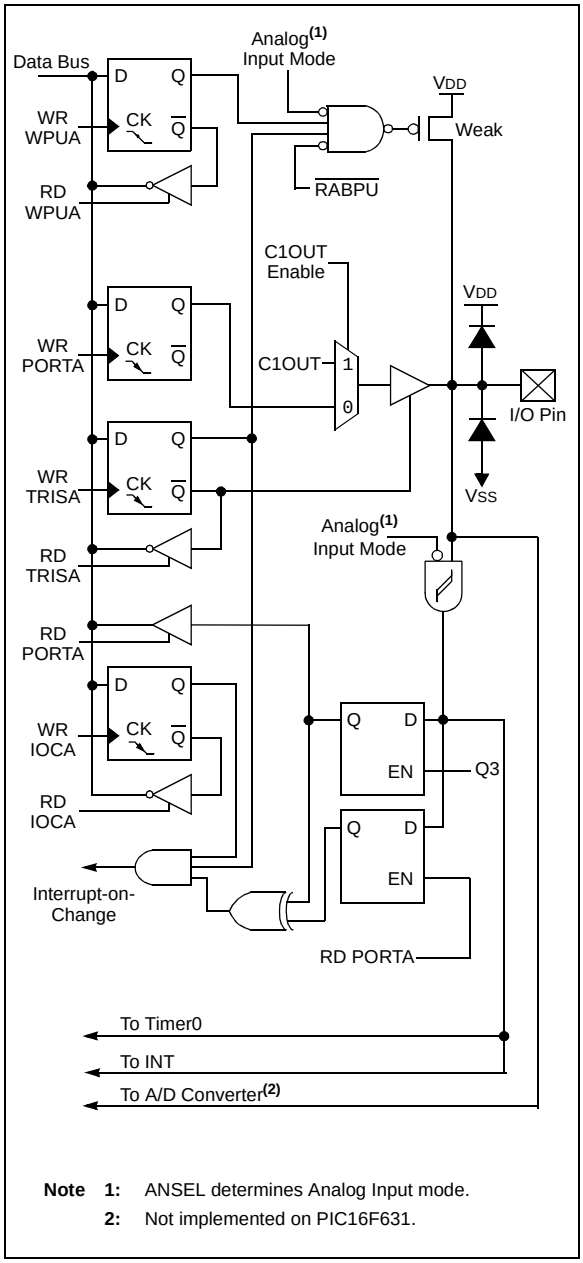
Note 1: ANSEL determines Analog Input mode.
2: Not implemented on PIC16F631.

4.2.5.3 RA2/AN2/T0CKI/INT/C1OUT

Figure 4-3 shows the diagram for this pin. The RA2/AN2/T0CKI/INT/C1OUT pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC (except PIC16F631)
- the clock input for Timer0
- an external edge triggered interrupt
- a digital output from Comparator C1

FIGURE 4-3: BLOCK DIAGRAM OF RA2



Note 1: ANSEL determines Analog Input mode.
2: Not implemented on PIC16F631.

8.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. The comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The Analog Comparator module includes the following features:

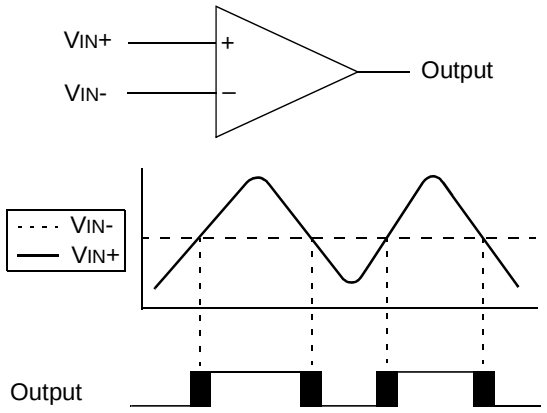
- Independent comparator control
- Programmable input selection
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- PWM shutdown
- Timer1 gate (count enable)
- Output synchronization to Timer1 clock input
- SR Latch
- Programmable and Fixed Voltage Reference

Note:	Only Comparator C2 can be linked to Timer1.

8.1 Comparator Overview

A single comparator is shown in Figure 8-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

FIGURE 8-1: SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

8.7 Analog Input Connection Considerations

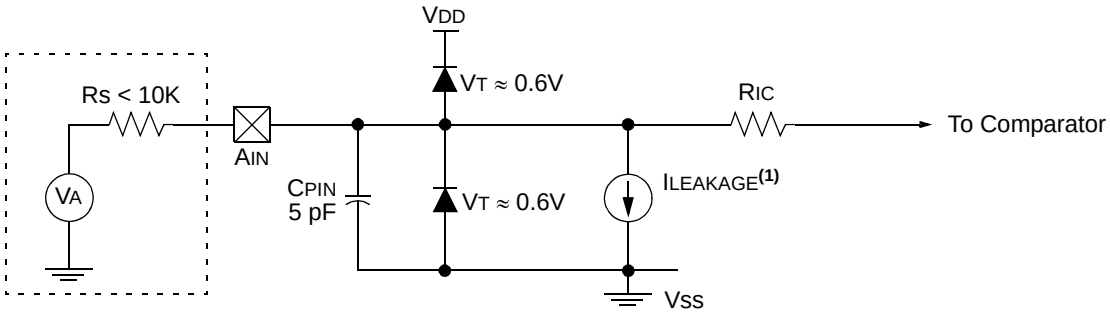
A simplified circuit for an analog input is shown in Figure 8-6. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 kΩ is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

Note 1: When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification.

2: Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

FIGURE 8-6: ANALOG INPUT MODEL



Legend: CPIN = Input Capacitance
ILEAKAGE = Leakage Current at the pin due to various junctions
RIC = Interconnect Resistance
RS = Source Impedance
VA = Analog Voltage
VT = Threshold Voltage

Note 1: See **Section 17.0 "Electrical Specifications"**

10.1.2 READING THE DATA EEPROM MEMORY

To read a data memory location, the user must write the address to the EEADR register, clear the EEPGD control bit of the EECON1 register, and then set control bit RD. The data is available at the very next cycle, in the EEDAT register; therefore, it can be read in the next instruction. EEDAT will hold this value until another read or until it is written to by the user (during a write operation).

EXAMPLE 10-1: DATA EEPROM READ

```
BANKSEL EEADR   ;
MOVF    DATA_EE_ADDR, W;
MOVWF   EEADR       ;Data Memory
                    ;Address to read
BANKSEL EECON1      ;
BCF     EECON1, EEPGD;Point to DATA memory
BSF     EECON1, RD  ;EE Read
BANKSEL EEDAT       ;
MOVF    EEDAT, W    ;W = EEDAT
BANKSEL PORTA       ;Bank 0
```

10.1.3 WRITING TO THE DATA EEPROM MEMORY

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDAT register. Then the user must follow a specific sequence to initiate the write for each byte.

The write will not initiate if the specific sequence is not followed exactly (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. Interrupts should be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

EXAMPLE 10-2: DATA EEPROM WRITE

```
        BANKSELEEADR    ;
        MOVFDATA_EE_ADDR, W;
        MOVWFEEADR      ;Data Memory Address to write
        MOVFDATA_EE_DATA, W;
        MOVWFEEDAT      ;Data Memory Value to write
        BANKSELEECON1   ;
        BCF EECON1, EEPGD;Point to DATA memory
        BSF EECON1, WREN;Enable writes

        BCF INTCON, GIE ;Disable INTs.
        BTFSCINTCON, GIE;SEE AN576
        GOTO$-2
Required MOVLW55h       ;
Sequence MOVWFEECON2    ;Write 55h
        MOVLWAAh        ;
        MOVWFEECON2     ;Write AAh
        BSF EECON1, WR  ;Set WR bit to begin write
        BSF INTCON, GIE ;Enable INTs.

        SLEEP           ;Wait for interrupt to signal write complete (optional)
        BCF EECON1, WREN;Disable writes
        BANKSEL0x00     ;Bank 0
```

10.1.4 READING THE FLASH PROGRAM MEMORY (PIC16F685/PIC16F689/PIC16F690)

To read a program memory location, the user must write the Least and Most Significant address bits to the EEADR and EEADRH registers, set the EEPGD control bit of the EECON1 register, and then set control bit RD. Once the read control bit is set, the program memory Flash controller will use the second instruction cycle to read the data. This causes the second instruction immediately following the "BSF EECON1,RD" instruction to be ignored. The data is available in the very next cycle, in the EEDAT and EEDATH registers; therefore, it can be read as two bytes in the following instructions.

EEDAT and EEDATH registers will hold this value until another read or until it is written to by the user.

Note 1:	The two instructions following a program memory read are required to be NOPs. This prevents the user from executing a 2-cycle instruction on the next instruction after the RD bit is set.
2:	If the WR bit is set when EEPGD = 1, it will be immediately reset to '0' and no operation will take place.

EXAMPLE 10-3: FLASH PROGRAM READ

```
    BANKSEL EEADR                   ;
    MOVF    MS_PROG_EE_ADDR, W      ;
    MOVWF   EEADRH                  ;MS Byte of Program Address to read
    MOVF    LS_PROG_EE_ADDR, W      ;
    MOVWF   EEADR                   ;LS Byte of Program Address to read
    BANKSELEECON1               ;
    BSF     EECON1, EEPGD           ;Point to PROGRAM memory
    BSF     EECON1, RD              ;EE Read
                                    ;
    NOP                             ;First instruction after BSF EECON1,RD executes normally
                                    ;
    NOP                             ;Any instructions here are ignored as program
                                    ;memory is read in second cycle after BSF EECON1,RD
;
    BANKSELEEDAT                ;
    MOVF    EEDAT, W                ;W = LS Byte of Program Memory
    MOVWF   LOWPMBYTE               ;
    MOVF    EEDATH, W               ;W = MS Byte of Program EEDAT
    MOVWF   HIGHPMBYTE              ;
    BANKSEL0x00                 ;Bank 0
```

Required Sequence

11.4.4 ENHANCED PWM AUTO-SHUTDOWN MODE

The PWM mode supports an Auto-Shutdown mode that will disable the PWM outputs when an external shutdown event occurs. Auto-Shutdown mode places the PWM output pins into a predetermined state. This mode is used to help prevent the PWM from damaging the application.

The auto-shutdown sources are selected using the ECCPASx bits of the ECCPAS register. A shutdown event may be generated by:

- A logic '0' on the INT pin
- Comparator C1
- Comparator C2
- Setting the ECCPASE bit in firmware

A shutdown condition is indicated by the ECCPASE (Auto-Shutdown Event Status) bit of the ECCPAS register. If the bit is a '0', the PWM pins are operating normally. If the bit is a '1', the PWM outputs are in the shutdown state.

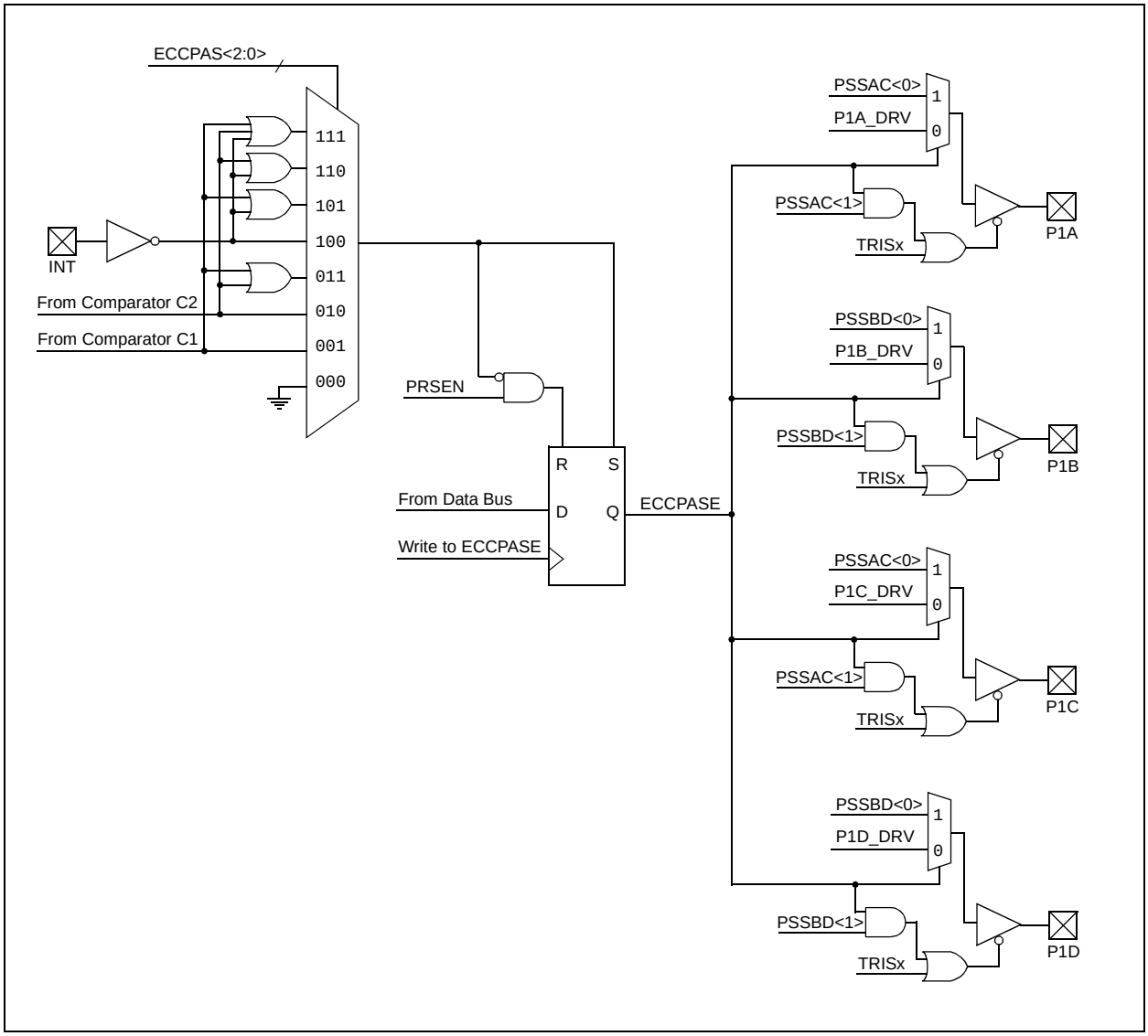
When a shutdown event occurs, two things happen:

The ECCPASE bit is set to '1'. The ECCPASE will remain set until cleared in firmware or an auto-restart occurs (see **Section 11.4.5 "Auto-Restart Mode"**).

The enabled PWM pins are asynchronously placed in their shutdown states. The PWM output pins are grouped into pairs [P1A/P1C] and [P1B/P1D]. The state of each pin pair is determined by the PSSAC and PSSBD bits of the ECCPAS register. Each pin pair may be placed into one of three states:

- Drive logic '1'
- Drive logic '0'
- Tri-state (high-impedance)

FIGURE 11-14: AUTO-SHUTDOWN BLOCK DIAGRAM



REGISTER 11-2: ECCPAS: ENHANCED CAPTURE/COMPARE/PWM AUTO-SHUTDOWN CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
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 7 **ECCPASE:** ECCP Auto-Shutdown Event Status bit
- 1 = A shutdown event has occurred; ECCP outputs are in shutdown state
- 0 = ECCP outputs are operating

bit 6-4 **ECCPAS<2:0>:** ECCP Auto-shutdown Source Select bits
- 000 =Auto-Shutdown is disabled
- 001 =Comparator C1 output high
- 010 =Comparator C2 output high(1)
- 011 =Either Comparators output is high
- 100 =VIL on INT pin
- 101 =VIL on INT pin or Comparator C1 output high
- 110 =VIL on INT pin or Comparator C2 output high
- 111 =VIL on INT pin or either Comparators output is high

bit 3-2 **PSSACn:** Pins P1A and P1C Shutdown State Control bits
- 00 = Drive pins P1A and P1C to '0'
- 01 = Drive pins P1A and P1C to '1'
- 1x = Pins P1A and P1C tri-state

bit 1-0 **PSSBDn:** Pins P1B and P1D Shutdown State Control bits
- 00 = Drive pins P1B and P1D to '0'
- 01 = Drive pins P1B and P1D to '1'
- 1x = Pins P1B and P1D tri-state

Note 1: If C2SYNC is enabled, the shutdown will be delayed by Timer1.

Note 1: The auto-shutdown condition is a level-based signal, not an edge-based signal. As long as the level is present, the auto-shutdown will persist.

2: Writing to the ECCPASE bit is disabled while an auto-shutdown condition persists.

3: Once the auto-shutdown condition has been removed and the PWM restarted (either through firmware or auto-restart) the PWM signal will always restart at the beginning of the next PWM period.

12.2 Clock Accuracy with Asynchronous Operation

The factory calibrates the internal oscillator block output (INTOSC). However, the INTOSC frequency may drift as VDD or temperature changes, and this directly affects the asynchronous baud rate. Two methods may be used to adjust the baud rate clock, but both require a reference clock source of some kind.

The first (preferred) method uses the OSCTUNE register to adjust the INTOSC output. Adjusting the value in the OSCTUNE register allows for fine resolution changes to the system clock source. See **Section 3.5 "Internal Clock Modes"** for more information.

The other method adjusts the value in the Baud Rate Generator. This can be done automatically with the Auto-Baud Detect feature (see **Section 12.3.1 "Auto-Baud Detect"**). There may not be fine enough resolution when adjusting the Baud Rate Generator to compensate for a gradual change in the peripheral clock frequency.

REGISTER 12-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN(1)	SYNC	SENDB	BRGH	TRMT	TX9D
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 7 **CSRC:** Clock Source Select bit
Asynchronous mode:
Don't care
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit enabled
0 = Transmit disabled

bit 4 **SYNC:** EUSART Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **SENDB:** Send Break Character bit
Asynchronous mode:
1 = Send Sync Break on next transmission (cleared by hardware upon completion)
0 = Sync Break transmission completed
Synchronous mode:
Don't care

bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 13-2: SSPCON: SYNC SERIAL PORT 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
WCOL	SSPOV	SSPEN	CKP	SSPM3(2)	SSPM2(2)	SSPM1(2)	SSPM0(2)
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 7 **WCOL**: Write Collision Detect bit
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6 **SSPOV**: Receive Overflow Indicator bit
In SPI mode:
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in Slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.
0 = No overflow
In I^2C™ mode:
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode. SSPOV must be cleared in software in either mode.
0 = No overflow

bit 5 **SSPEN**: Synchronous Serial Port Enable bit
In SPI mode:
1 = Enables serial port and configures SCK, SDO and SDI as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In I^2C mode:
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In both modes, when enabled, these pins must be properly configured as input or output.

bit 4 **CKP**: Clock Polarity Select bit
In SPI mode:
1 = Idle state for clock is a high level (Microwire default)
0 = Idle state for clock is a low level (Microwire alternate)
In I^2C mode:
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

bit 3-0 **SSPM<3:0>**: Synchronous Serial Port Mode Select bits
0000 = SPI Master mode, clock = FOSC/4
0001 = SPI Master mode, clock = FOSC/16
0010 = SPI Master mode, clock = FOSC/64
0011 = SPI Master mode, clock = TMR2 output/2
0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.
0110 = I^2C Slave mode, 7-bit address
0111 = I^2C Slave mode, 10-bit address
1000 = Reserved
1001 = Load SSPMSK register at SSPADD SFR address(2)
1010 = Reserved
1011 = I^2C Firmware Controlled Master mode (slave IDLE)
1100 = Reserved
1101 = Reserved
1110 = I^2C Slave mode, 7-bit address with Start and Stop bit interrupts enabled
1111 = I^2C Slave mode, 10-bit address with Start and Stop bit interrupts enabled

Note 1: PIC16F687/PIC16F689/PIC16F690 only.
2: When this mode is selected, any reads or writes to the SSPADD SFR address actually accesses the SSPMSK register.

13.12.3 SSP MASK REGISTER

An SSP Mask (SSPMSK) register is available in I^2C Slave mode as a mask for the value held in the SSPSR register during an address comparison operation. A zero (‘0’) bit in the SSPMSK register has the effect of making the corresponding bit in the SSPSR register a ‘don’t care’.

This register is reset to all ‘1’s upon any Reset condition and, therefore, has no effect on standard SSP operation until written with a mask value.

This register must be initiated prior to setting SSPM<3:0> bits to select the I^2C Slave mode (7-bit or 10-bit address).

This register can only be accessed when the appropriate mode is selected by bits (SSPM<3:0> of SSPCON).

The SSP Mask register is active during:

- 7-bit Address mode: address compare of A<7:1>.
- 10-bit Address mode: address compare of A<7:0> only. The SSP mask has no effect during the reception of the first (high) byte of the address.

REGISTER 13-3: SSPMSK: SSP MASK REGISTER(1)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0(2)
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 7-1 **MSK<7:1>:** Mask bits

1 = The received address bit n is compared to SSPADD<n> to detect I^2C address match
0 = The received address bit n is not used to detect I^2C address match

bit 0 **MSK<0>:** Mask bit for I^2C Slave mode, 10-bit Address(2)

I^2C Slave mode, 10-bit Address (SSPM<3:0> = 0111):
1 = The received address bit 0 is compared to SSPADD<0> to detect I^2C address match
0 = The received address bit 0 is not used to detect I^2C address match

Note 1: When SSPCON bits SSPM<3:0> = 1001, any reads or writes to the SSPADD SFR address are accessed through the SSPMSK register. The SSPEN bit of the SSPCON register should be zero when accessing the SSPMSK register.

2: In all other SSP modes, this bit has no effect.

DECFSZ	Decrement f, Skip if 0
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 → (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a 2-cycle instruction.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \le k \le 2047$
Operation:	k → PC<10:0> PCLATH<4:3> → PC<12:11>
Status Affected:	None
Description:	GOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a 2-cycle instruction.

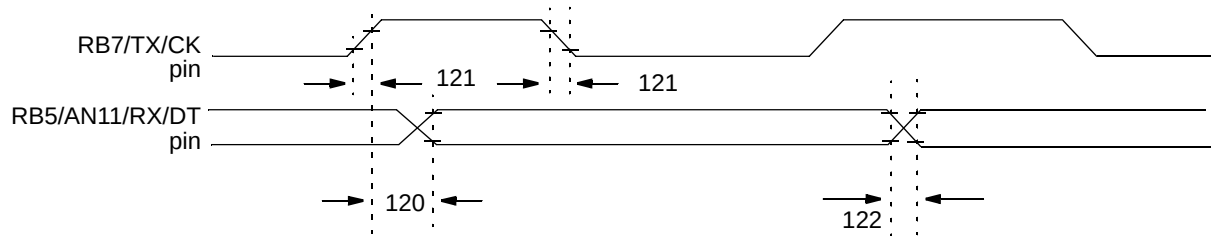
INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

INCFSZ	Increment f, Skip if 0
Syntax:	[*label*] INCFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 → (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a 2-cycle instruction.

IORLW	Inclusive OR literal with W
Syntax:	[*label*] IORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .OR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the 8-bit literal 'k'. The result is placed in the W register.

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .OR. (f) → (destination)
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

FIGURE 17-10: EUSART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

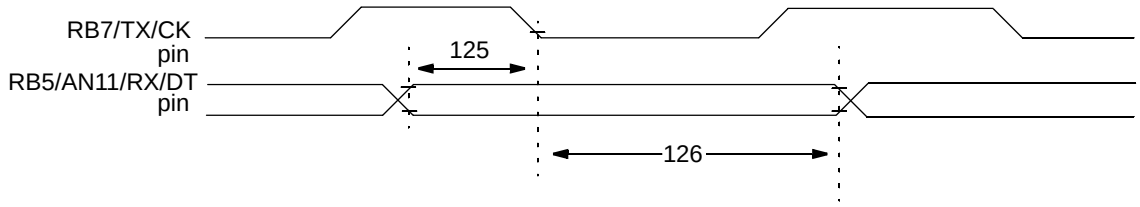


Note: Refer to Figure 17-3 for load conditions.

TABLE 17-10: EUSART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
120	TCKH2DTV	SYNC XMIT (Master & Slave) Clock high to data-out valid	—	40	ns	
121	TCKRF	Clock out rise time and fall time (Master mode)	—	20	ns	
122	TDTRF	Data-out rise time and fall time	—	20	ns	

FIGURE 17-11: EUSART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



Note: Refer to Figure 17-3 for load conditions.

TABLE 17-11: EUSART SYNCHRONOUS RECEIVE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
125	TDTV2CKL	SYNC RCV (Master & Slave) Data-hold before CK ↓ (DT hold time)	10	—	ns	
126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	—	ns	

FIGURE 18-14: MAXIMUM IPD VS. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED)

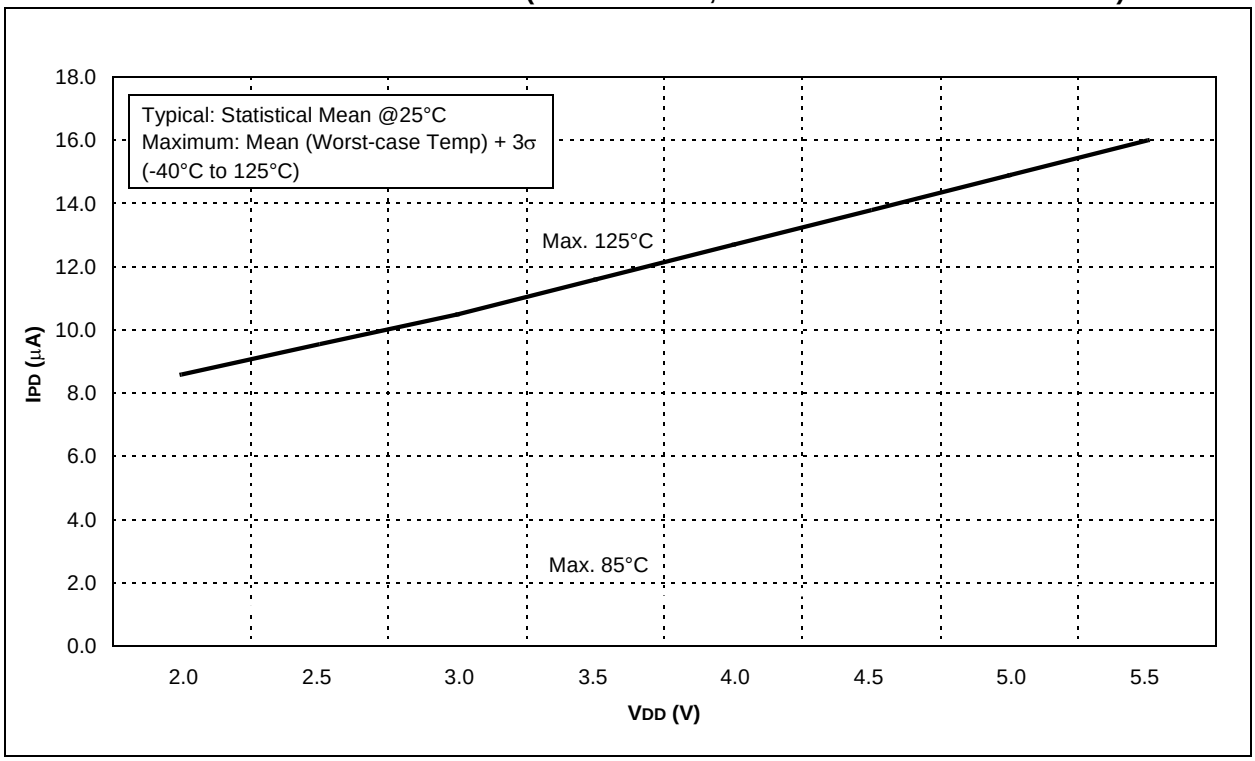


FIGURE 18-15: COMPARATOR IPD VS. VDD (BOTH COMPARATORS ENABLED)

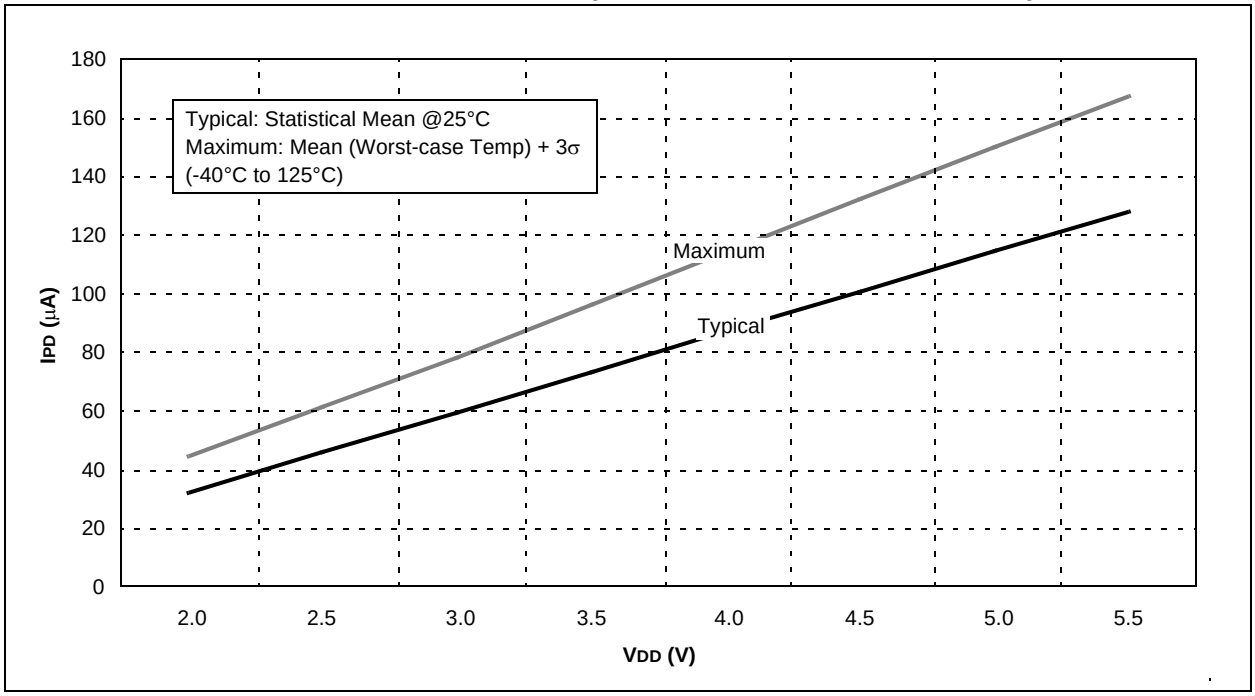


FIGURE 18-26: VOL VS. IOL OVER TEMPERATURE (VDD = 3.0V)

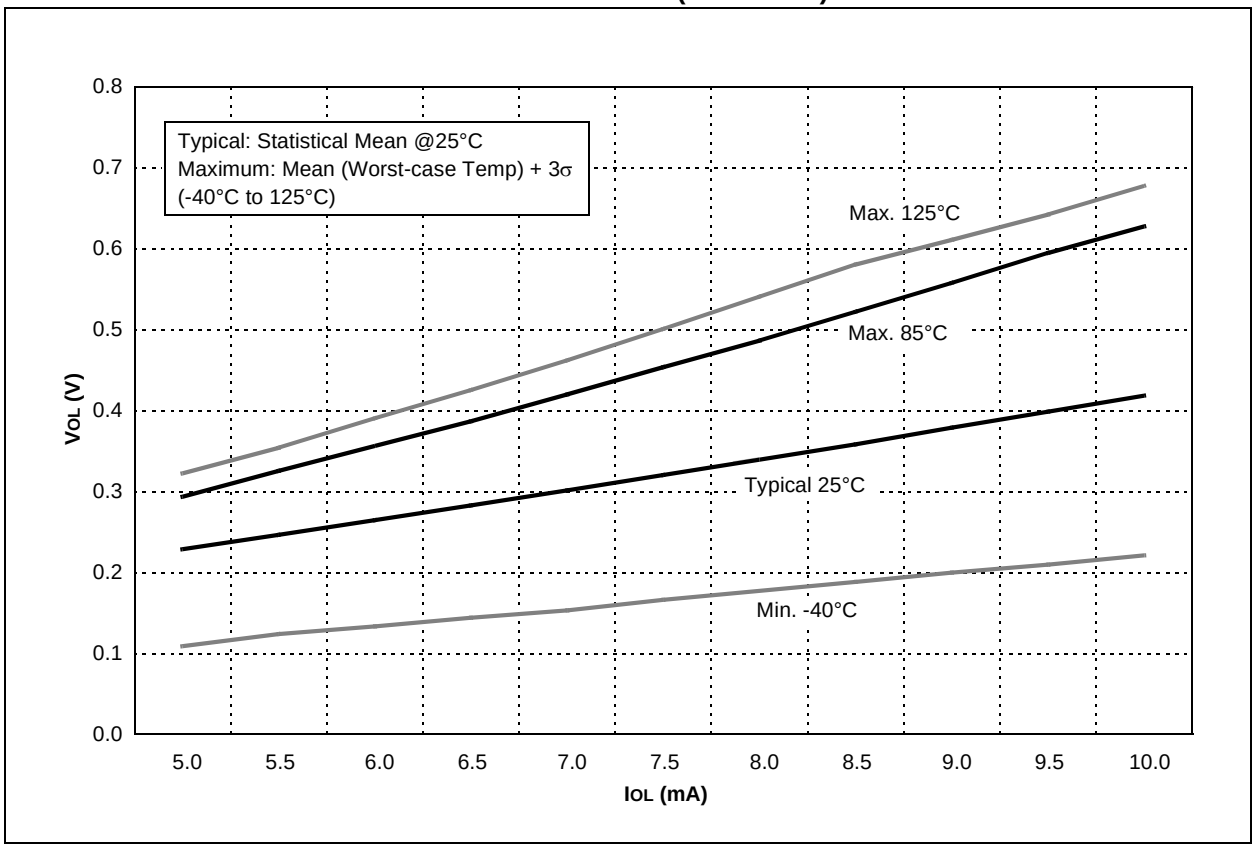
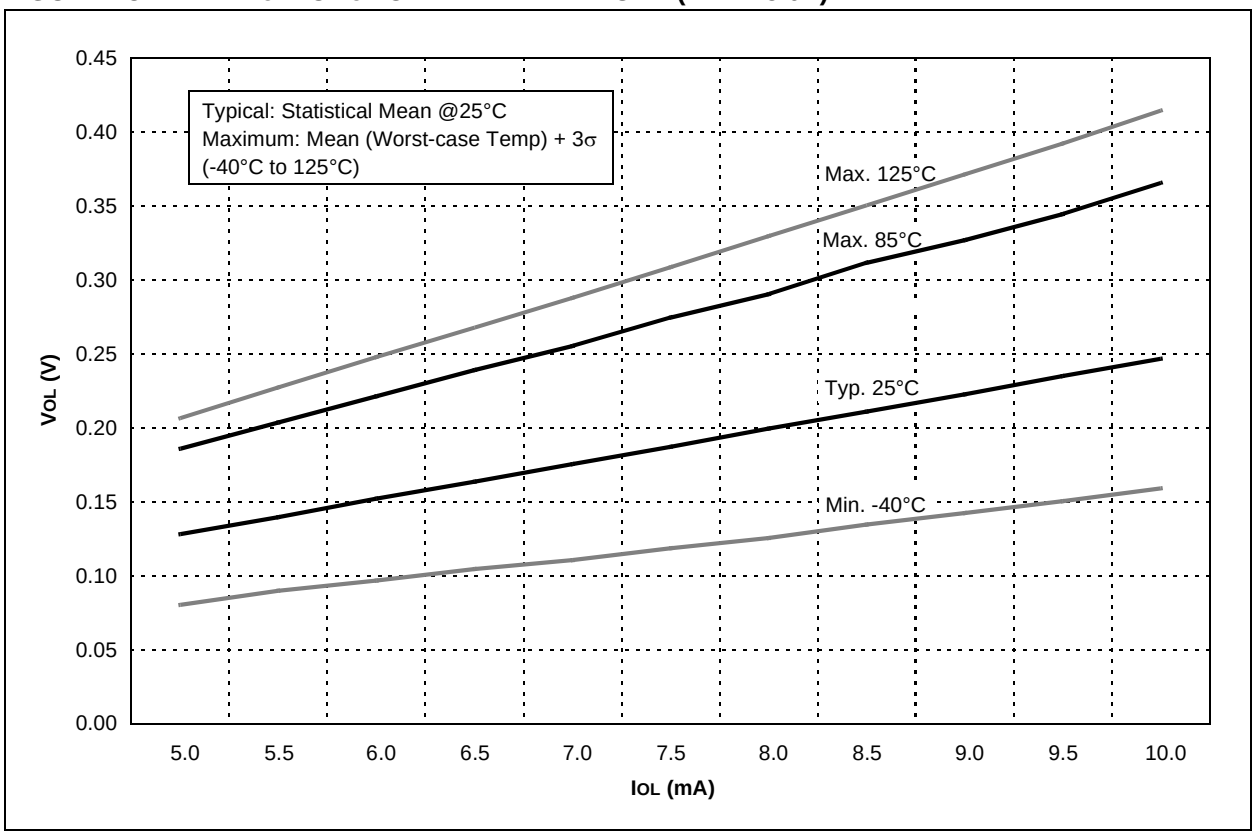
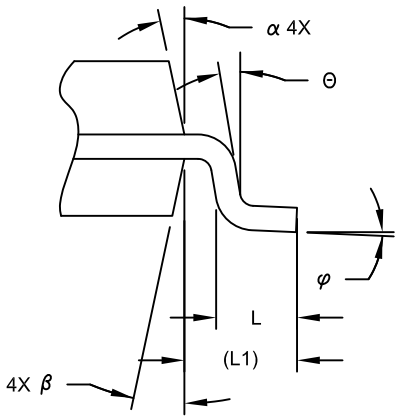


FIGURE 18-27: VOL VS. IOL OVER TEMPERATURE (VDD = 5.0V)

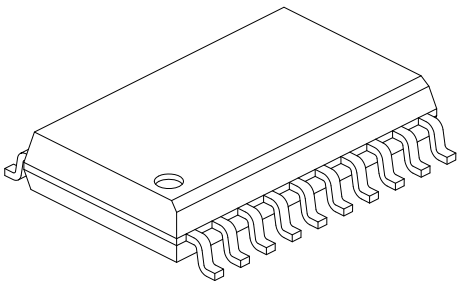


20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



VIEW C



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		1.27 BSC	
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		12.80 BSC	
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1		1.40 REF	
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.20	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

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

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 of 2

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (March 2005)

This is a new data sheet.

Revision B (May 2006)

Added 631/677 part numbers; Added pin summary tables after pin diagrams; Incorporated Golden Chapters.

Revision C (July 2006)

Revised Section 4.2.1, ANSEL and ANSELH Registers; Register 4-3, ANSEL Analog Select; Added Register 4-4, ANSELH Analog Select High; Section 11.3.2, Revised CCP1<1:0> to DC1B<1:0>; Section 11.3.7, Number 4 - Revised CCP1 to DC1B; Figure 11-5, Revised CCP1 to DC1B; Table 11-4, Revised P1M to P1M<1:0>; Section 12.3.1, Revised Paragraph 3; Revised Note 2; Revised Figure 12-6 Title.

Revision D (February 2007)

Removed Preliminary status; Changed PICmicro to PIC; Replaced Dev. Tool Section; Replaced Package Drawings.

Revision E (March 2008)

Add Char Data charts; Updated EUSART Golden Chapter; Updated the Electrical Specification section; Updated Package Drawings as needed.

Revision F (April 2015)

Added Section 17.8: High Temperature Operation in the Electrical Specifications section.

APPENDIX B: MIGRATING FROM OTHER PIC® DEVICES

This discusses some of the issues in migrating from other PIC devices to the PIC16F6XX Family of devices.

B.1 PIC16F676 to PIC16F685

TABLE B-1: FEATURE COMPARISON

Feature	PIC16F676	PIC16F685
Max Operating Speed	20 MHz	20 MHz
Max Program Memory (Words)	1024	4096
SRAM (bytes)	64	128
A/D Resolution	10-bit	10-bit
Data EEPROM (Bytes)	128	256
Timers (8/16-bit)	1/1	2/1
Oscillator Modes	8	8
Brown-out Reset	Y	Y
Internal Pull-ups	RA0/1/2/4/5	RA0/1/2/4/5, $\overline{\text{MCLR}}$
Interrupt-on-change	RA0/1/2/3/4/5	RA0/1/2/3/4/5
Comparator	1	2
ECCP+	N	Y
Ultra Low-Power Wake-up	N	Y
Extended WDT	N	Y
Software Control Option of WDT/BOR	N	Y
INTOSC Frequencies	4 MHz	31 kHz-8 MHz
Clock Switching	N	Y

Note: This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.