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	64MHz
Connectivity	ECANbus, I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.6K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f26k80-i-ss

NOTES:

© 2010-2012 Microchip Technology Inc.

REGISTER 3-1: OSCCON: OSCILLATOR CONTROL REGISTER (CONTINUED)

bit 2 **HFIOFS:** HF-INTOSC Frequency Stable bit
1 = HF-INTOSC oscillator frequency is stable
0 = HF-INTOSC oscillator frequency is not stable

bit 1-0 **SCS<1:0>:** System Clock Select bits(4)
1x = Internal oscillator block (LF-INTOSC, MF-INTOSC or HF-INTOSC)
01 = SOSC oscillator
00 = Default primary oscillator (OSC1/OSC2 or HF-INTOSC with or without PLL; defined by the FOSC<3:0> Configuration bits, CONFIG1H<3:0>)

Note 1: The Reset state depends on the state of the IESO Configuration bit (CONFIG1H<7>).
2: Modifying these bits will cause an immediate clock frequency switch if the internal oscillator is providing the device clocks.
3: The source is selected by the INTSRC bit (OSCTUNE<7>).
4: Modifying these bits will cause an immediate clock source switch.
5: INTSRC = OSCTUNE<7> and MFIOSEL = OSCCON2<0>.
6: This is the lowest power option for an internal source.

REGISTER 3-2: OSCCON2: OSCILLATOR CONTROL REGISTER 2

U-0	R-0	U-0	R/W-1	R/W-0	U-0	R-x	R/W-0
—	SOSCRUN	—	SOSCDRV(1)	SOSCGO	—	MFIOFS	MFIOSEL
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 **Unimplemented:** Read as '0'

bit 6 **SOSCRUN:** SOSC Run Status bit
1 = System clock comes from a secondary SOSC
0 = System clock comes from an oscillator other than SOSC

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

bit 4 **SOSCDRV:** Secondary Oscillator Drive Control bit(1)
1 = High-power SOSC circuit is selected
0 = Low/high-power select is done via the SOSCSEL<1:0> Configuration bits

bit 3 **SOSCGO:** Oscillator Start Control bit
1 = Oscillator is running even if no other sources are requesting it.
0 = Oscillator is shut off if no other sources are requesting it (When the SOSC is selected to run from a digital clock input, rather than an external crystal, this bit has no effect.)

bit 2 **Unimplemented:** Read as '0'

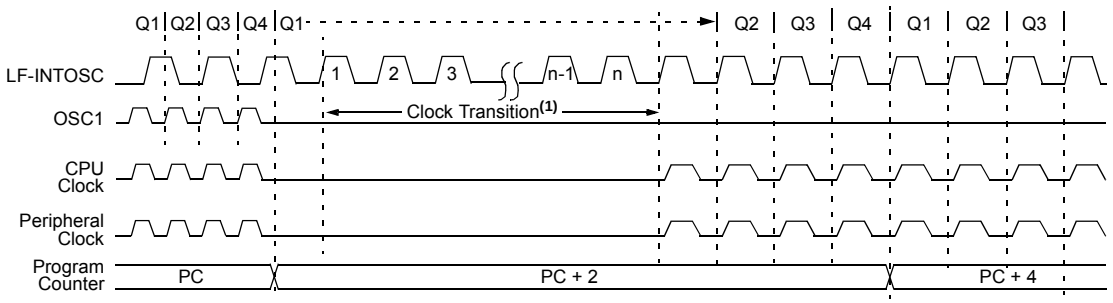
bit 1 **MFIOFS:** MF-INTOSC Frequency Stable bit
1 = MF-INTOSC is stable
0 = MF-INTOSC is not stable

bit 0 **MFIOSEL:** MF-INTOSC Select bit
1 = MF-INTOSC is used in place of HF-INTOSC frequencies of 500 kHz, 250 kHz and 31.25 kHz
0 = MF-INTOSC is not used

Note 1: When SOSC is selected to run from a digital clock input, rather than an external crystal, this bit has no effect.

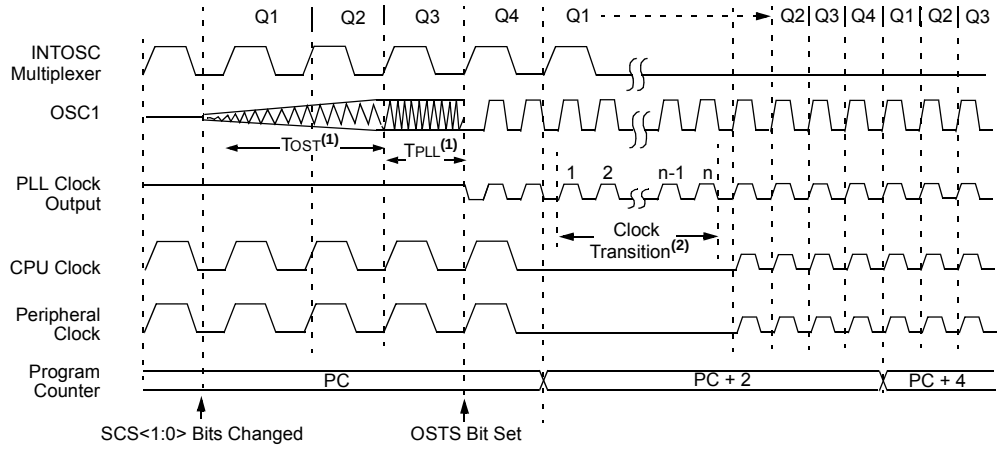
© 2010-2012 Microchip Technology Inc.

FIGURE 4-3: TRANSITION TIMING TO RC_RUN MODE



Note 1: Clock transition typically occurs within 2-4 TOSC.

FIGURE 4-4: TRANSITION TIMING FROM RC_RUN MODE TO PRI_RUN MODE



Note 1: TOST = 1024 TOSC; TPLL = 2 ms (approx). These intervals are not shown to scale.

2: Clock transition typically occurs within 2-4 TOSC.

REGISTER 7-1: EECON1: EEPROM CONTROL REGISTER 1

R/W-x	R/W-x	U-0	R/W-0	R/W-x	R/W-0	R/S-0	R/S-0
EEPGD	CFGS	—	FREE	WRERR(1)	WREN	WR	RD
bit 7							bit 0

Legend:	S = Settable 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 7 **EEPGD:** Flash Program or Data EEPROM Memory Select bit
- 1 = Accesses Flash program memory
- 0 = Accesses data EEPROM memory

bit 6 **CFGS:** Flash Program/Data EEPROM or Configuration Select bit
- 1 = Accesses Configuration registers
- 0 = Accesses Flash program or data EEPROM memory

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

bit 4 **FREE:** Flash Row Erase Enable bit
- 1 = Erases the program memory row addressed by TBLPTR on the next WR command (cleared by completion of erase operation)
- 0 = Performs write-only

bit 3 **WRERR:** Flash Program/Data EEPROM Error Flag bit(1)
- 1 = A write operation is prematurely terminated (any Reset during self-timed programming in normal operation or an improper write attempt)
- 0 = The write operation completed

bit 2 **WREN:** Flash Program/Data EEPROM Write Enable bit
- 1 = Allows write cycles to Flash program/data EEPROM
- 0 = Inhibits write cycles to Flash program/data EEPROM

bit 1 **WR:** Write Control bit
- 1 = Initiates a data EEPROM erase/write cycle or, a program memory erase cycle or write cycle. (The operation is self-timed and the bit is cleared by hardware once the write is complete. The WR bit can only be set (not cleared) in software.)
- 0 = Write cycle to the EEPROM is complete

bit 0 **RD:** Read Control bit
- 1 = Initiates an EEPROM read (Read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software. RD bit cannot be set when EEPGD = 1 or CFGS = 1.)
- 0 = Does not initiate an EEPROM read

Note 1: When a WRERR occurs, the EEPGD and CFGS bits are not cleared. This allows tracing of the error condition.

EXAMPLE 7-3: WRITING TO FLASH PROGRAM MEMORY (CONTINUED)

```
PROGRAM_MEMORY
                BSF     EECON1, EEPGD       ; point to Flash program memory
                BCF     EECON1, CFGS        ; access Flash program memory
                BSF     EECON1, WREN        ; enable write to memory
                BCF     INTCON, GIE         ; disable interrupts
                MOVLW   55h
    Required    MOVWF   EECON2              ; write 55h
    Sequence    MOVLW   0AAh
                MOVWF   EECON2              ; write 0AAh
                BSF     EECON1, WR          ; start program (CPU stall)
                BSF     INTCON, GIE         ; re-enable interrupts
                BCF     EECON1, WREN        ; disable write to memory
```

7.5.2 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

7.5.3 UNEXPECTED TERMINATION OF WRITE OPERATION

If a write is terminated by an unplanned event, such as loss of power or an unexpected Reset, the memory location just programmed should be verified and reprogrammed if needed. If the write operation is interrupted by a $\overline{\text{MCLR}}$ Reset or a WDT Time-out Reset during normal operation, the user can check the WRERR bit and rewrite the location(s) as needed.

7.5.4 PROTECTION AGAINST SPURIOUS WRITES

To protect against spurious writes to Flash program memory, the write initiate sequence must also be followed. See **Section 28.0 "Special Features of the CPU"** for more detail.

7.6 Flash Program Operation During Code Protection

See **Section 28.6 "Program Verification and Code Protection"** for details on code protection of Flash program memory.

TABLE 7-2: REGISTERS ASSOCIATED WITH PROGRAM FLASH MEMORY

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TBLPTRU	—	—	bit 21(1)	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)				
TBPLTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)							
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)							
TABLAT	Program Memory Table Latch							
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
EECON2	EEPROM Control Register 2 (not a physical register)							
EECON1	EEPGD	CFGS	—	FREE	WRERR	WREN	WR	RD
IPR4	TMR4IP	EEIP	CMP2IP	CMP1IP	—	CCP5IP	CCP4IP	CCP3IP
PIR4	TMR4IF	EEIF	CMP2IF	CMP1IF	—	CCP5IF	CCP4IF	CCP3IF
PIE4	TMR4IE	EEIE	CMP2IE	CMP1IE	—	CCP5IE	CCP4IE	CCP3IE

Legend: — = unimplemented, read as '0'. Shaded cells are not used during Flash/EEPROM access.

Note 1: Bit 21 of the TBLPTRU allows access to the device Configuration bits.

11.4 PORTC, TRISC and LATC Registers

PORTC is an eight-bit wide, bidirectional port. The corresponding Data Direction and Output Latch registers are TRISC and LATC. Only PORTC pins, RC2 through RC7, are digital only pins.

PORTC is multiplexed with CCP, MSSP and EUSARTx peripheral functions (Table 11-5). The pins have Schmitt Trigger input buffers. The pins for CCP, SPI and EUSARTx are also configurable for open-drain output whenever these functions are active. Open-drain configuration is selected by setting the SSPOD, CCPxOD and U1OD control bits in the ODCON register.

RC1 is configurable for open-drain output when CCP2 is active on this pin. Open-drain configuration is selected by setting the CCP2OD control bit (ODCON<3>).

When enabling peripheral functions, use care in defining TRIS bits for each PORTC pin. Some peripherals can override the TRIS bit to make a pin an output or input. Consult the corresponding peripheral section for the correct TRIS bit settings.

Note:	These pins are configured as digital inputs on any device Reset.

The contents of the TRISC register are affected by peripheral overrides. Reading TRISC always returns the current contents, even though a peripheral device may be overriding one or more of the pins.

EXAMPLE 11-3: INITIALIZING PORTC

```
CLRF    PORTC   ; Initialize PORTC by
                ; clearing output
                ; data latches
CLRF    LATC    ; Alternate method
                ; to clear output
                ; data latches
MOVLW   0CFh    ; Value used to
                ; initialize data
                ; direction
MOVWF   TRISC   ; Set RC<3:0> as inputs
                ; RC<5:4> as outputs
                ; RC<7:6> as inputs
```

11.6 PORTE, TRISE and LATE Registers

PORTE is a seven-bit-wide, bidirectional port. The corresponding Data Direction and Output Latch registers are TRISE and LATE.

Note: PORTE is unavailable on 28-pin devices.

All pins on PORTE are implemented with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

Note: These pins are configured as digital inputs on any device Reset.

Each of the PORTE pins has a weak internal pull-up. A single control bit can turn off all the pull-ups. This is performed by clearing bit, REPU (PADCFG1<6>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on any device Reset.

PORTE is also multiplexed with the Parallel Slave Port address lines. RE1 and RE0 are multiplexed with the Parallel Slave Port (PSP) control signals, $\overline{WR}$ and $\overline{RD}$.

EXAMPLE 11-5: INITIALIZING PORTE

```
CLRF    PORTE   ; Initialize PORTE by
                ; clearing output
                ; data latches
CLRF    LATE    ; Alternate method
                ; to clear output
                ; data latches
MOVLW   03h     ; Value used to
                ; initialize data
                ; direction
MOVWF   TRISE   ; Set RE<1:0> as inputs
                ; RE<7:2> as outputs
```

TABLE 11-9: PORTE FUNCTIONS

Pin Name	Function	TRIS Setting	I/O	I/O Type	Description
RE0/AN5/$\overline{RD}$	RE0	0	O	DIG	LATE<0> data output.
		1	I	ST	PORTE<0> data input.
	AN5	1	I	ANA	A/D Input Channel 5. Default input configuration on POR; does not affect digital output.
	$\overline{RD}$	x	O	DIG	Parallel Slave Port read strobe pin.
		x	I	ST	Parallel Slave Port read pin.
RE1/AN6/ C1OUT/$\overline{WR}$	RE1	0	O	DIG	LATE<1> data output.
		1	I	ST	PORTE<1> data input.
	AN6	1	I	ANA	A/D Input Channel 5. Default input configuration on POR; does not affect digital output.
	C1OUT	0	O	DIG	Comparator 1 output; takes priority over port data.
	$\overline{WR}$	x	O	DIG	Parallel Slave Port write strobe pin.
		x	I	ST	Parallel Slave Port write pin.
RE2/AN7/ C2OUT/$\overline{CS}$	RE2	0	O	DIG	LATE<2> data output.
		1	I	ST	PORTE<2> data input.
	AN7	1	I	ANA	A/D Input Channel 7. Default input configuration on POR; does not affect digital output.
	C2OUT	0	O	DIG	Comparator 2 output; takes priority over port data.
	$\overline{CS}$	x	I	ST	Parallel Slave Port chip select.
RE3	RE3	1	I	ST	PORT<3> data input.
RE4/CANRX	RE4(1)	0	O	DIG	LATE<4> data output.
		1	I	ST	PORTE<4> data input.
	CANRX(1,2)	1	I	ST	CAN bus RX.

Legend: O = Output, I = Input, ANA = Analog Signal, DIG = CMOS Output, ST = Schmitt Trigger Buffer Input, x = Don't care (TRIS bit does not affect port direction or is overridden for this option)

Note 1: These bits are unavailable for 40 and 44-pin devices (PIC18F4XK0).

2: This is the alternate pin assignment for CANRX and CANTX on 64-pin devices (PIC18F6XK80) when the CANMX Configuration bit is cleared.

REGISTER 12-1: MDCON: MODULATION CONTROL REGISTER

R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	U-0	U-0	R/W-0
MDEN	MDOE	MDSLR	MDOPOL	MDO	—	—	MDBIT
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 **MDEN:** Modulator Module Enable bit
1 = Modulator module is enabled and mixing input signals
0 = Modulator module is disabled and has no output

bit 6 **MDOE:** Modulator Module Pin Output Enable bit
1 = Modulator pin output is enabled
0 = Modulator pin output is disabled

bit 5 **MDSLR:** MDOUT Pin Slew Rate Limiting bit
1 = MDOUT pin slew rate limiting is enabled
0 = MDOUT pin slew rate limiting is disabled

bit 4 **MDOPOL:** Modulator Output Polarity Select bit
1 = Modulator output signal is inverted
0 = Modulator output signal is not inverted

bit 3 **MDO:** Modulator Output bit
Displays the current output value of the modulator module.**(2)**

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **MDBIT:** Modulator Source Input bit
Allows software to manually set modulation source input to module.**(1)**

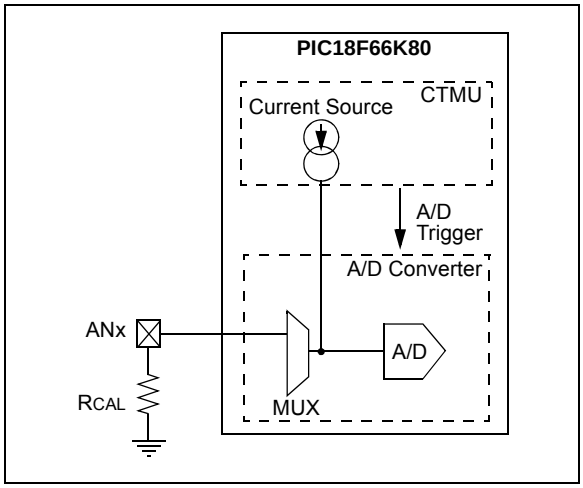
Note 1: The MDBIT must be selected as the modulation source in the MDSRC register for this operation.
2: The modulated output frequency can be greater and asynchronous from the clock that updates this register bit. The bit value may not be valid for higher speed modulator or carrier signals.

The CTMU current source may be trimmed with the trim bits in CTMUICON, using an iterative process to get the exact current desired. Alternatively, the nominal value without adjustment may be used. That value may be stored by software, for use in all subsequent capacitive or time measurements.

To calculate the optimal value for RCAL, the nominal current must be chosen.

For example, if the A/D Converter reference voltage is 3.3V, use 70% of full scale (or 2.31V) as the desired approximate voltage to be read by the A/D Converter. If the range of the CTMU current source is selected to be 0.55 μA, the resistor value needed is calculated as RCAL = 2.31V/0.55 μA, for a value of 4.2 MΩ. Similarly, if the current source is chosen to be 5.5 μA, RCAL would be 420,000Ω, and 42,000Ω if the current source is set to 55 μA.

FIGURE 18-2: CTMU CURRENT SOURCE CALIBRATION CIRCUIT



A value of 70% of full-scale voltage is chosen to make sure that the A/D Converter is in a range that is well above the noise floor. If an exact current is chosen to incorporate the trimming bits from CTMUICON, the resistor value of RCAL may need to be adjusted accordingly. RCAL also may be adjusted to allow for available resistor values. RCAL should be of the highest precision available, in light of the precision needed for the circuit that the CTMU will be measuring. A recommended minimum would be 0.1% tolerance.

The following examples show a typical method for performing a CTMU current calibration.

- Example 18-1 demonstrates how to initialize the A/D Converter and the CTMU.

  This routine is typical for applications using both modules.

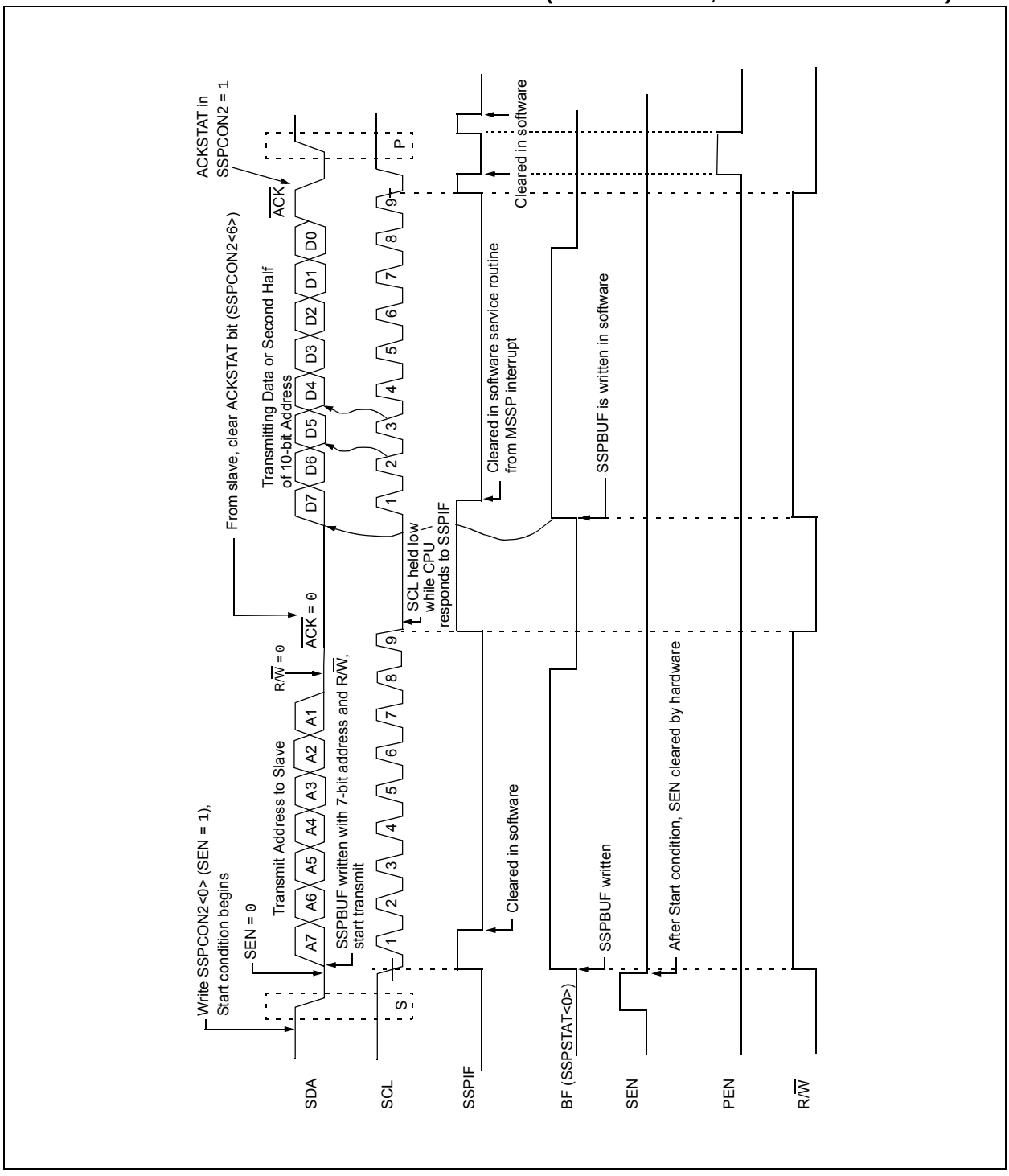
- Example 18-2 demonstrates one method for the actual calibration routine.

  This method manually triggers the A/D Converter to demonstrate the entire step-wise process. It is also possible to automatically trigger the conversion by setting the CTMU's CTTRIG bit (CTMUCONH<0>).

EXAMPLE 18-4: ROUTINE FOR CAPACITIVE TOUCH SWITCH

```
#include "p18cxxx.h"

#define COUNT 500                       //@ 8MHz = 125uS.
#define DELAY for(i=0;i<COUNT;i++)
#define OPENSW 1000                     //Un-pressed switch value
#define TRIP 300                        //Difference between pressed
                                        //and un-pressed switch
#define HYST 65                         //amount to change
                                        //from pressed to un-pressed
#define PRESSED 1
#define UNPRESSED 0

int main(void)
{
    unsigned int Vread;                 //storage for reading
    unsigned int switchState;
    int i;

    //assume CTMU and A/D have been setup correctly
    //see Example 25-1 for CTMU & A/D setup
    setup();

    CTMUCONHbits.CTMUEN = 1;            //Enable the CTMU

    CTMUCONHbits.IDISSEN = 1;           //drain charge on the circuit
    DELAY;                              //wait 125us
    CTMUCONHbits.IDISSEN = 0;           //end drain of circuit

    CTMUCONLbits.EDG1STAT = 1;          //Begin charging the circuit
                                        //using CTMU current source
    DELAY;                              //wait for 125us
    CTMUCONLbits.EDG1STAT = 0;          //Stop charging circuit

    PIR1bits.ADIF = 0;                  //make sure A/D Int not set
    ADCON0bits.GO=1;                    //and begin A/D conv.
    while(!PIR1bits.ADIF);              //Wait for A/D convert complete

    Vread = ADRES;                      //Get the value from the A/D

    if(Vread < OPENSW - TRIP)
    {
        switchState = PRESSED;
    }
    else if(Vread > OPENSW - TRIP + HYST)
    {
        switchState = UNPRESSED;
    }
}
```

FIGURE 21-23: I²C™ MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



25.0 COMPARATOR VOLTAGE REFERENCE MODULE

The comparator voltage reference is a 32-tap resistor ladder network that provides a selectable reference voltage. Although its primary purpose is to provide a reference for the analog comparators, it may also be used independently of them.

A block diagram of the module is shown in Figure 25-1. The resistor ladder is segmented to provide a range of CVREF values and has a power-down function to conserve power when the reference is not being used. The module's supply reference can be provided from either device VDD/VSS or an external voltage reference.

25.1 Configuring the Comparator Voltage Reference

The comparator voltage reference module is controlled through the CVRCON register (Register 25-1). The comparator voltage reference provides a range of output voltage with 32 levels.

The CVR<4:0> selection bits (CVRCON<4:0>) offer a range of output voltages. Equation 25-1 shows the how the comparator voltage reference is computed.

EQUATION 25-1:

If CVRSS = 1:

$$CV_{REF} = \left(V_{REF-} + \frac{CVR<4:0>}{32}\right) \bullet (V_{REF+} - V_{REF-})$$

If CVRSS = 0:

$$CV_{REF} = \left(AV_{SS} + \frac{CVR<4:0>}{32}\right) \bullet (AV_{DD} - AV_{SS})$$

The comparator reference supply voltage can come from either VDD and VSS, or the external VREF+ and VREF- that are multiplexed with RA3 and RA2. The voltage source is selected by the CVRSS bit (CVRCON<5>).

The settling time of the comparator voltage reference must be considered when changing the CVREF output (see Table 31-2 in **Section 31.0 "Electrical Characteristics"**).

REGISTER 25-1: CVRCON: COMPARATOR VOLTAGE REFERENCE 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
CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0
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 **CVREN:** Comparator Voltage Reference Enable bit
1 = CVREF circuit powered on
0 = CVREF circuit powered down

bit 6 **CVROE:** Comparator VREF Output Enable bit
1 = CVREF voltage level is output on CVREF pin
0 = CVREF voltage level is disconnected from CVREF pin

bit 5 **CVRSS:** Comparator VREF Source Selection bit
1 = Comparator reference source, CVRSRC = VREF+ – VREF-
0 = Comparator reference source, CVRSRC = AVDD – AVSS

bit 4-0 **CVR<4:0>:** Comparator VREF Value Selection 0 ≤ CVR<4:0> ≤ 31 bits
When CVRSS = 1:
CVREF = (VREF-) + (CVR<4:0>/32) • (VREF+ – VREF-)
When CVRSS = 0:
CVREF = (AVSS) + (CVR<4:0>/32) • (AVDD – AVSS)

27.4.3 MODE 2 – ENHANCED FIFO MODE

In Mode 2, two or more receive buffers are used to form the receive FIFO (first in, first out) buffer. There is no one-to-one relationship between the receive buffer and acceptance filter registers. Any filter that is enabled and linked to any FIFO receive buffer can generate acceptance and cause FIFO to be updated.

FIFO length is user-programmable, from 2-8 buffers deep. FIFO length is determined by the very first programmable buffer that is configured as a transmit buffer. For example, if Buffer 2 (B2) is programmed as a transmit buffer, FIFO consists of RXB0, RXB1, B0 and B1, creating a FIFO length of 4. If all programmable buffers are configured as receive buffers, FIFO will have the maximum length of 8.

The following is the list of resources available in Mode 2:

- Three transmit buffers: TXB0, TXB1 and TXB2
- Two receive buffers: RXB0 and RXB1
- Six buffers programmable as TX or RX; receive buffers form FIFO: B0-B5
- Automatic RTR handling on B0-B5
- Sixteen acceptance filters: RXF0-RXF15
- Two dedicated acceptance mask registers; RXF15 programmable as third mask: RXM0-RXM1, RXF15
- Programmable data filter on standard identifier messages: SDFLC, useful for DeviceNet protocol

27.5 CAN Message Buffers

27.5.1 DEDICATED TRANSMIT BUFFERS

The PIC18F66K80 family devices implement three dedicated transmit buffers – TXB0, TXB1 and TXB2. Each of these buffers occupies 14 bytes of SRAM and are mapped into the SFR memory map. These are the only transmit buffers available in Mode 0. Mode 1 and 2 may access these and other additional buffers.

Each transmit buffer contains one Control register (TXBnCON), four Identifier registers (TXBnSIDL, TXBnSIDH, TXBnEIDL, TXBnEIDH), one Data Length Count register (TXBnDLC) and eight Data Byte registers (TXBnDm).

27.5.2 DEDICATED RECEIVE BUFFERS

The PIC18F66K80 family devices implement two dedicated receive buffers: RXB0 and RXB1. Each of these buffers occupies 14 bytes of SRAM and are mapped into SFR memory map. These are the only receive buffers available in Mode 0. Mode 1 and 2 may access these and other additional buffers.

Each receive buffer contains one Control register (RXBnCON), four Identifier registers (RXBnSIDL, RXBnSIDH, RXBnEIDL, RXBnEIDH), one Data Length Count register (RXBnDLC) and eight Data Byte registers (RXBnDm).

There is also a separate Message Assembly Buffer (MAB) which acts as an additional receive buffer. MAB is always committed to receiving the next message from the bus and is not directly accessible to user firmware. The MAB assembles all incoming messages one by one. A message is transferred to appropriate receive buffers only if the corresponding acceptance filter criteria is met.

27.5.3 PROGRAMMABLE TRANSMIT/ RECEIVE BUFFERS

The ECAN module implements six new buffers: B0-B5. These buffers are individually programmable as either transmit or receive buffers. These buffers are available only in Mode 1 and 2. As with dedicated transmit and receive buffers, each of these programmable buffers occupies 14 bytes of SRAM and are mapped into SFR memory map.

Each buffer contains one Control register (BnCON), four Identifier registers (BnSIDL, BnSIDH, BnEIDL, BnEIDH), one Data Length Count register (BnDLC) and eight Data Byte registers (BnDm). Each of these registers contains two sets of control bits. Depending on whether the buffer is configured as transmit or receive, one would use the corresponding control bit set. By default, all buffers are configured as receive buffers. Each buffer can be individually configured as a transmit or receive buffer by setting the corresponding TXENn bit in the BSEL0 register.

When configured as transmit buffers, user firmware may access transmit buffers in any order similar to accessing dedicated transmit buffers. In receive configuration with Mode 1 enabled, user firmware may also access receive buffers in any order required. But in Mode 2, all receive buffers are combined to form a single FIFO. Actual FIFO length is programmable by user firmware. Access to FIFO must be done through the FIFO Pointer bits (FP<4:0>) in the CANCON register. It must be noted that there is no hardware protection against out of order FIFO reads.

© 2010-2012 Microchip Technology Inc.

27.5.4 PROGRAMMABLE AUTO-RTR BUFFERS

In Mode 1 and 2, any of six programmable transmit/receive buffers may be programmed to automatically respond to predefined RTR messages without user firmware intervention. Automatic RTR handling is enabled by setting the TX2EN bit in the BSEL0 register and the RTREN bit in the BnCON register. After this setup, when an RTR request is received, the TXREQ bit is automatically set and the current buffer content is automatically queued for transmission as a RTR response. As with all transmit buffers, once the TXREQ bit is set, buffer registers become read-only and any writes to them will be ignored.

The following outlines the steps required to automatically handle RTR messages:

1. Set buffer to Transmit mode by setting the TXnEN bit to '1' in the BSEL0 register.
2. At least one acceptance filter must be associated with this buffer and preloaded with the expected RTR identifier.
3. Bit, RTREN in the BnCON register, must be set to '1'.
4. Buffer must be preloaded with the data to be sent as a RTR response.

Normally, user firmware will keep buffer data registers up to date. If firmware attempts to update the buffer while an automatic RTR response is in the process of transmission, all writes to buffers are ignored.

27.6 CAN Message Transmission

27.6.1 INITIATING TRANSMISSION

For the MCU to have write access to the message buffer, the TXREQ bit must be clear, indicating that the message buffer is clear of any pending message to be transmitted. At a minimum, the SIDH, SIDL and DLC registers must be loaded. If data bytes are present in the message, the Data registers must also be loaded. If the message is to use extended identifiers, the EIDH:EIDL registers must also be loaded and the EXIDE bit set.

To initiate message transmission, the TXREQ bit must be set for each buffer to be transmitted. When TXREQ is set, the TXABT, TXLARB and TXERR bits will be cleared. To successfully complete the transmission, there must be at least one node with matching baud rate on the network.

Setting the TXREQ bit does not initiate a message transmission; it merely flags a message buffer as ready for transmission. Transmission will start when the device detects that the bus is available. The device will then begin transmission of the highest priority message that is ready.

When the transmission has completed successfully, the TXREQ bit will be cleared, the TXBnIF bit will be set and an interrupt will be generated if the TXBnIE bit is set.

If the message transmission fails, the TXREQ will remain set, indicating that the message is still pending for transmission and one of the following condition flags will be set. If the message started to transmit but encountered an error condition, the TXERR and the IRXIF bits will be set and an interrupt will be generated. If the message lost arbitration, the TXLARB bit will be set.

27.6.2 ABORTING TRANSMISSION

The MCU can request to abort a message by clearing the TXREQ bit associated with the corresponding message buffer (TXBnCON<3> or BnCON<3>). Setting the ABAT bit (CANCON<4>) will request an abort of all pending messages. If the message has not yet started transmission, or if the message started but is interrupted by loss of arbitration or an error, the abort will be processed. The abort is indicated when the module sets the TXABT bit for the corresponding buffer (TXBnCON<6> or BnCON<6>). If the message has started to transmit, it will attempt to transmit the current message fully. If the current message is transmitted fully and is not lost to arbitration or an error, the TXABT bit will not be set because the message was transmitted successfully. Likewise, if a message is being transmitted during an abort request and the message is lost to arbitration or an error, the message will not be retransmitted and the TXABT bit will be set, indicating that the message was successfully aborted.

Once an abort is requested by setting the ABAT or TXABT bits, it cannot be cleared to cancel the abort request. Only CAN module hardware or a POR condition can clear it.

28.4 Two-Speed Start-up

The Two-Speed Start-up feature helps to minimize the latency period from oscillator start-up to code execution by allowing the microcontroller to use the INTOSC (LF-INTOSC, MF-INTOSC, HF-INTOSC) oscillator as a clock source until the primary clock source is available. It is enabled by setting the IESO Configuration bit.

Two-Speed Start-up should be enabled only if the primary oscillator mode is LP, XT or HS (Crystal-Based modes). Other sources do not require an OST start-up delay; for these, Two-Speed Start-up should be disabled.

When enabled, Resets and wake-ups from Sleep mode cause the device to configure itself to run from the internal oscillator block as the clock source, following the time-out of the Power-up Timer after a Power-on Reset is enabled. This allows almost immediate code execution while the primary oscillator starts and the OST is running. Once the OST times out, the device automatically switches to PRI_RUN mode.

To use a higher clock speed on wake-up, the INTOSC or postscaler clock sources can be selected to provide a higher clock speed by setting bits, IRCF<2:0>, immediately after Reset. For wake-ups from Sleep, the INTOSC or postscaler clock sources can be selected by setting the IRCF2:0> bits prior to entering Sleep mode.

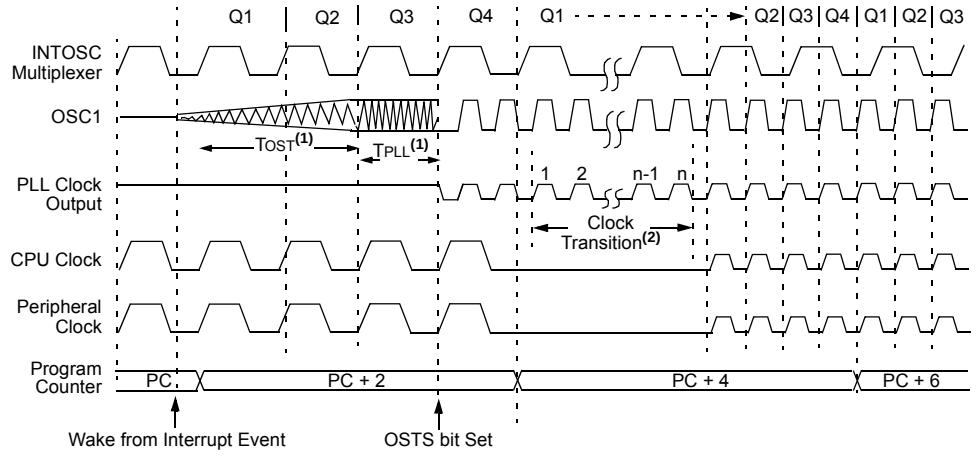
In all other power-managed modes, Two-Speed Start-up is not used. The device will be clocked by the currently selected clock source until the primary clock source becomes available. The setting of the IESO bit is ignored.

28.4.1 SPECIAL CONSIDERATIONS FOR USING TWO-SPEED START-UP

While using the INTOSC oscillator in Two-Speed Start-up, the device still obeys the normal command sequences for entering power-managed modes, including multiple SLEEP instructions (refer to **Section 4.1.4 "Multiple Sleep Commands"**). In practice, this means that user code can change the SCS<1:0> bit settings or issue SLEEP instructions before the OST times out. This would allow an application to briefly wake-up, perform routine "housekeeping" tasks and return to Sleep before the device starts to operate from the primary oscillator.

User code can also check if the primary clock source is currently providing the device clocking by checking the status of the OSTS bit (OSCCON<3>). If the bit is set, the primary oscillator is providing the clock. Otherwise, the internal oscillator block is providing the clock during wake-up from Reset or Sleep mode.

FIGURE 28-3: TIMING TRANSITION FOR TWO-SPEED START-UP (INTOSC TO HSPLL)



Note 1: TOST = 1024 TOSC; TPLL = 2 ms (approx). These intervals are not shown to scale.

2: Clock transition typically occurs within 2-4 TOSC.

© 2010-2012 Microchip Technology Inc.

28.6.2 DATA EEPROM CODE PROTECTION

The entire data EEPROM is protected from external reads and writes by two bits: CPD and WRTD. CPD inhibits external reads and writes of data EEPROM. WRTD inhibits internal and external writes to data EEPROM. The CPU can always read data EEPROM under normal operation, regardless of the protection bit settings.

28.6.3 CONFIGURATION REGISTER PROTECTION

The Configuration registers can be write-protected. The WRTC bit controls protection of the Configuration registers. In normal execution mode, the WRTC bit is readable only. WRTC can only be written via ICSP or an external programmer.

28.7 ID Locations

Eight memory locations (200000h-200007h) are designated as ID locations, where the user can store checksum or other code identification numbers. These locations are both readable and writable during normal execution through the TBLRD and TBLWT instructions or during program/verify. The ID locations can be read when the device is code-protected.

28.8 In-Circuit Serial Programming

The PIC18F66K80 family of devices can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data and three other lines for power, ground and the programming voltage. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

For the various programming modes, see the programming specification

28.9 In-Circuit Debugger

When the $\overline{\text{DEBUG}}$ Configuration bit is programmed to a '0', the In-Circuit Debugger functionality is enabled. This function allows simple debugging functions when used with MPLAB® IDE. When the microcontroller has this feature enabled, some resources are not available for general use. Table 28-5 shows which resources are required by the background debugger.

TABLE 28-5: DEBUGGER RESOURCES

I/O Pins:	RB6, RB7
Stack:	Two levels
Program Memory:	512 bytes
Data Memory:	10 bytes

To use the In-Circuit Debugger function of the microcontroller, the design must implement In-Circuit Serial Programming connections to $\overline{\text{MCLR}}$/RE3, VDD, VSS, RB7 and RB6. This will interface to the In-Circuit Debugger module available from Microchip or one of the third-party development tool companies.

© 2010-2012 Microchip Technology Inc.

BNC **Branch if Not Carry**

Syntax: BNC n

Operands: -128 ≤ n ≤ 127

Operation: if Carry bit is '0',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0011	nnnn	nnnn

Description: If the Carry bit is '0', then the program will branch.

The 2's complement number, '2n', is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNC Jump

Before Instruction
PC = address (HERE)
After Instruction
If Carry = 0;
PC = address (Jump)
If Carry = 1;
PC = address (HERE + 2)

BNN **Branch if Not Negative**

Syntax: BNN n

Operands: -128 ≤ n ≤ 127

Operation: if Negative bit is '0',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0111	nnnn	nnnn

Description: If the Negative bit is '0', then the program will branch.

The 2's complement number, '2n', is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNN Jump

Before Instruction
PC = address (HERE)
After Instruction
If Negative = 0;
PC = address (Jump)
If Negative = 1;
PC = address (HERE + 2)

31.2 DC Characteristics: Power-Down and Supply Current PIC18F66K80 Family (Industrial/Extended) (Continued)

PIC18F66K80 Family (Industrial/Extended)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD) Cont.(2,3)						
	PIC18**LF**XXK80	260	380	µA	-40°C	VDD = 1.8V(4) Regulator Disabled	FOSC = 1 MHz (**RC_IDLE** mode, HF-INTOSC)
		260	380	µA	+25°C		
		260	380	µA	+60°C		
		270	390	µA	+85°C		
		280	420	µA	+125°C		
	PIC18**LF**XXK80	400	500	µA	-40°C	VDD = 3.3V(4) Regulator Disabled	
		400	500	µA	+25°C		
		400	500	µA	+60°C		
		410	520	µA	+85°C		
		420	580	µA	+125°C		
	PIC18**F**XXK80	430	560	µA	-40°C	VDD = 3.3V(5) Regulator Enabled	
		430	560	µA	+25°C		
		430	560	µA	+60°C		
		450	580	µA	+85°C		
		480	620	µA	+125°C		
	PIC18**F**XXK80	450	620	µA	-40°C	VDD = 5V(5) Regulator Enabled	
		450	620	µA	+25°C		
		450	620	µA	+60°C		
		470	640	µA	+85°C		
		500	680	µA	+125°C		

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;

$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: For LF devices, $\overline{RETEN}$ (CONFIG1L<0>) = 1.

5: For F devices, SRETEN (WDTCON<4>) = 1 and $\overline{RETEN}$ (CONFIG1L<0>) = 0.

31.2 DC Characteristics: Power-Down and Supply Current PIC18F66K80 Family (Industrial/Extended) (Continued)

PIC18F66K80 Family (Industrial/Extended)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD) Cont.(2,3)						
	PIC18**LF**XXK80	270	600	µA	-40°C	VDD = 1.8V(4) Regulator Disabled	FOSC = 4 MHz (**PRI_RUN** mode, EC oscillator)
		270	600	µA	+25°C		
		270	600	µA	+60°C		
		300	700	µA	+85°C		
		320	850	µA	+125°C		
	PIC18**LF**XXK80	540	1000	µA	-40°C	VDD = 3.3V(4) Regulator Disabled	
		540	1000	µA	+25°C		
		540	1000	µA	+60°C		
		550	1100	µA	+85°C		
		560	1200	µA	+125°C		
	PIC18**F**XXK80	566	1020	µA	-40°C	VDD = 3.3V(5) Regulator Enabled	
		585	1020	µA	+25°C		
		590	1020	µA	+60°C		
		595	1120	µA	+85°C		
		600	1220	µA	+125°C		
	PIC18**F**XXK80	630	2000	µA	-40°C	VDD = 5V(5) Regulator Enabled	
		630	2000	µA	+25°C		
		630	2000	µA	+60°C		
		640	2000	µA	+85°C		
		650	2000	µA	+125°C		

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;

$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: For LF devices, $\overline{RETEN}$ (CONFIG1L<0>) = 1.

5: For F devices, SRETEN (WDTCON<4>) = 1 and $\overline{RETEN}$ (CONFIG1L<0>) = 0.

© 2010-2012 Microchip Technology Inc.

31.2 DC Characteristics: Power-Down and Supply Current PIC18F66K80 Family (Industrial/Extended) (Continued)

PIC18F66K80 Family (Industrial/Extended)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Module Differential Currents (ΔIWDT, ΔIBOR, ΔIHLVD, ΔIADC)						
D022 (ΔIWDT)	**Watchdog Timer**						
	PIC18**LF**XXK80	0.4	2	µA	-40°C to +125°C	VDD = 1.8V Regulator Disabled	
	PIC18**LF**XXK80	0.6	3	µA	-40°C to +125°C	VDD = 3.3V Regulator Disabled	
	PIC18**F**XXK80	0.6	3	µA	-40°C to +125°C	VDD = 3.3V Regulator Enabled	
	PIC18**F**XXK80	0.8	4	µA	-40°C to +125°C	VDD = 5.5V Regulator Enabled	
D022A (ΔIBOR)	**Brown-out Reset**						
	PIC18**LF**XXK80	4.6	20	µA	-40°C to +125°C	VDD = 1.8V Regulator Disabled	High-Power BOR
	PIC18**F**XXK80	4.6	20	µA	-40°C to +125°C	VDD = 3.3V Regulator Enabled	
	PIC18**F**XXK80	4.6	20	µA	-40°C to +125°C	VDD = 5.5V Regulator Enabled	
D022B ΔIHLVD	**High/Low-Voltage Detect**						
	PIC18**LF**XXK80	3.8	10	µA	-40°C to +125°C	VDD = 1.8V Regulator Disabled	
	PIC18L**F**XXK80	4.5	12	µA	-40°C to +125°C	VDD = 3.3V Regulator Disabled	
	PIC18**F**XXK80	3.8	12	µA	-40°C to +125°C	VDD = 3.3V Regulator Enabled	
	PIC18**F**XXK80	4.9	13	µA	-40°C to +125°C	VDD = 5.5V Regulator Enabled	
D026 ΔIADC	**A/D Converter**						
	PIC18**LF**XXK80	0.4	1.5		-40°C to +125°C	VDD = 1.8V Regulator Disabled	A/D on, not converting
	PIC18L**F**XXK80	0.5	2	µA	-40°C to +125°C	VDD = 3.3V Regulator Disabled	
	PIC18**F**XXK80	0.5	3	µA	-40°C to +125°C	VDD = 3.3V Regulator Enabled	
	PIC18**F**XXK80	1	3	µA	-40°C to +125°C	VDD = 5.5V Regulator Enabled	

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: For LF devices, $\overline{RETEN}$ (CONFIG1L<0>) = 1.

5: For F devices, SRETEN (WDTCON<4>) = 1 and $\overline{RETEN}$ (CONFIG1L<0>) = 0.

© 2010-2012 Microchip Technology Inc.