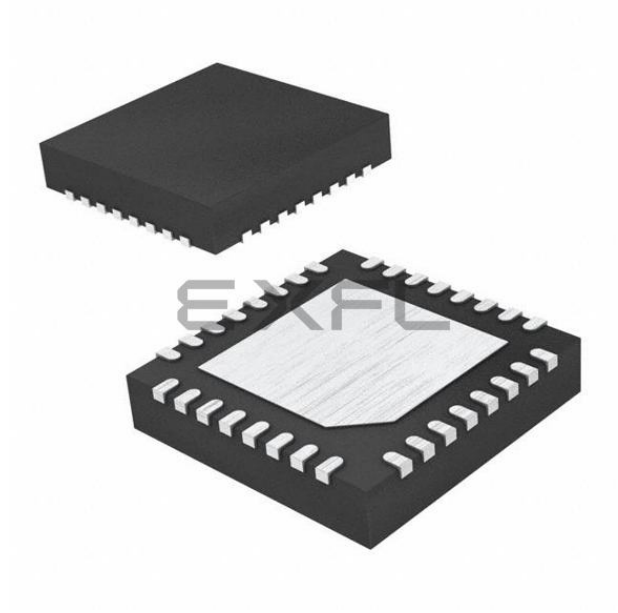


Welcome to E-XFL.COM

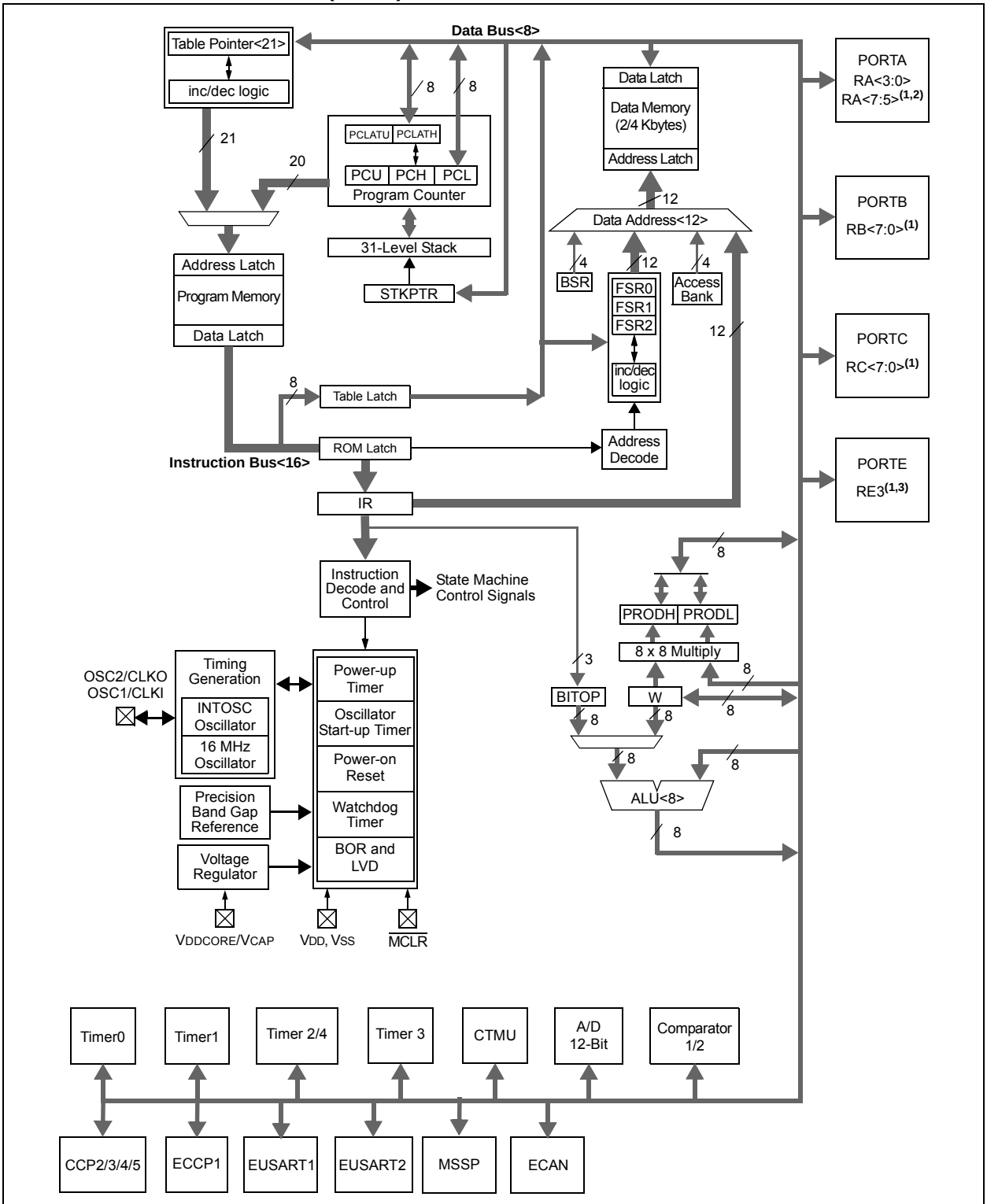
What is "Embedded - Microcontrollers"?

"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-VQFN Exposed Pad
Supplier Device Package	28-QFN-S (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f26k80-i-mm

FIGURE 1-1: PIC18F2XK80 (28-PIN) BLOCK DIAGRAM



Note 1: See Table 1-4 for I/O port pin descriptions.

2: RA6 and RA7 are only available as digital I/O in select oscillator modes. For more information, see **Section 3.0 "Oscillator Configurations"**.

3: RE3 is only available when the MCLRE Configuration bit is cleared (MCLRE = 0).

TABLE 1-6: PIC18F6XK80 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Num	Pin Type	Buffer Type	Description
$\overline{\text{MCLR}}$/RE3	28			
$\overline{\text{MCLR}}$		I	ST	Master Clear (input) or programming voltage (input). This pin is an active-low Reset to the device.
RE3		I	ST	General purpose, input only pin.
OSC1/CLKIN/RA7	46			
OSC1		I	ST	Oscillator crystal input.
CLKIN		I	CMOS	External clock source input. Always associated with pin function, OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.)
RA7		I/O	ST/ CMOS	General purpose I/O pin.
OSC2/CLKOUT/RA6	47			
OSC2		O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKOUT		O	—	In certain oscillator modes, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
RA6		I/O	ST/ CMOS	General purpose I/O pin.

Legend: I^2C™ = I^2C/SMBus input buffer
ST = Schmitt Trigger input with CMOS levels
I = Input
P = Power
CMOS = CMOS compatible input or output
Analog = Analog input
O = Output

3.3 Clock Sources and Oscillator Switching

Essentially, PIC18F66K80 family devices have these independent clock sources:

- Primary oscillators
- Secondary oscillators
- Internal oscillator

The **primary oscillators** can be thought of as the main device oscillators. These are any external oscillators connected to the OSC1 and OSC2 pins, and include the External Crystal and Resonator modes and the External Clock modes. If selected by the FOSC<3:0> Configuration bits (CONFIG1H<3:0>), the internal oscillator block may be considered a primary oscillator. The internal oscillator block can be one of the following:

- 31 kHz LF-INTOSC source
- 31 kHz to 500 kHz MF-INTOSC source
- 31 kHz to 16 MHz HF-INTOSC source

The particular mode is defined by the FOSCx Configuration bits. The details of these modes are covered in **Section 3.5 "External Oscillator Modes"**.

The **secondary oscillators** are external clock sources that are not connected to the OSC1 or OSC2 pin. These sources may continue to operate, even after the controller is placed in a power-managed mode. PIC18F66K80 family devices offer the SOSC (Timer1/3/5/7) oscillator as a secondary oscillator source.

The SOSC can be enabled from any peripheral that requests it. The SOSC can be enabled several ways by doing one of the following:

- The SOSC is selected as the source by either of the odd timers, which is done by each respective SOSCEN bit (TxCON<3>)
- The SOSC is selected as the CPU clock source by the SCSx bits (OSCCON<1:0>)
- The SOSCGO bit is set (OSCCON2<3>)

The SOSCGO bit is used to warm up the SOSC so that it is ready before any peripheral requests it.

The secondary oscillator has three Run modes. The SOSCSEL<1:0> bits (CONFIG1L<4:3>) decide the SOSC mode of operation:

- 11 = High-Power SOSC Circuit
- 10 = Digital (SCLKI) mode
- 11 = Low-Power SOSC Circuit

If a secondary oscillator is not desired and digital I/O on port pins, RC0 and RC1, is needed, the SOSCSELx bits must be set to Digital mode.

In addition to being a primary clock source in some circumstances, the **internal oscillator** is available as a power-managed mode clock source. The LF-INTOSC source is also used as the clock source for several special features, such as the WDT and Fail-Safe Clock Monitor. The internal oscillator block is discussed in more detail in **Section 3.6 "Internal Oscillator Block"**.

The PIC18F66K80 family includes features that allow the device clock source to be switched from the main oscillator, chosen by device configuration, to one of the alternate clock sources. When an alternate clock source is enabled, various power-managed operating modes are available.

3.3.1 OSC1/OSC2 OSCILLATOR

The OSC1/OSC2 oscillator block is used to provide the oscillator modes and frequency ranges:

Mode	Design Operating Frequency
LP	31.25-100 kHz
XT	100 kHz to 4 MHz
HS	4 MHz to 25 MHz
EC	0 to 64 MHz (external clock)
EXTRC	0 to 4 MHz (external RC)

The crystal-based oscillators (XT, HS and LP) have a built-in start-up time. The operation of the EC and EXTRC clocks is immediate.

3.3.2 CLOCK SOURCE SELECTION

The System Clock Select bits, SCS<1:0> (OSCCON<1:0>), select the clock source. The available clock sources are the primary clock defined by the FOSC<3:0> Configuration bits, the secondary clock (SOSC oscillator) and the internal oscillator. The clock source changes after one or more of the bits is written to, following a brief clock transition interval.

The OSTS (OSCCON<3>) and SOSCRUN (OSCCON2<6>) bits indicate which clock source is currently providing the device clock. The OSTS bit indicates that the Oscillator Start-up Timer (OST) has timed out and the primary clock is providing the device clock in primary clock modes. The SOSCRUN bit indicates when the SOSC oscillator (from Timer1/3/5/7) is providing the device clock in secondary clock modes. In power-managed modes, only one of these bits will be set at any time. If neither of these bits is set, the INTOSC is providing the clock or the internal oscillator has just started and is not yet stable.

The IDLEN bit (OSCCON<7>) determines if the device goes into Sleep mode or one of the Idle modes when the SLEEP instruction is executed.

REGISTER 3-4: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL(1)	RODIV3	RODIV2	RODIV1	RODIV0
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 **ROON:** Reference Oscillator Output Enable bit
1 = Reference oscillator output is available on REFO pin
0 = Reference oscillator output is disabled

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

bit 5 **ROSSLP:** Reference Oscillator Output Stop in Sleep bit
1 = Reference oscillator continues to run in Sleep
0 = Reference oscillator is disabled in Sleep

bit 4 **ROSEL:** Reference Oscillator Source Select bit(1)
1 = Primary oscillator (EC or HS) is used as the base clock
0 = System clock is used as the base clock; base clock reflects any clock switching of the device

bit 3-0 **RODIV<3:0>:** Reference Oscillator Divisor Select bits
1111 = Base clock value divided by 32,768
1110 = Base clock value divided by 16,384
1101 = Base clock value divided by 8,192
1100 = Base clock value divided by 4,096
1011 = Base clock value divided by 2,048
1010 = Base clock value divided by 1,024
1001 = Base clock value divided by 512
1000 = Base clock value divided by 256
0111 = Base clock value divided by 128
0110 = Base clock value divided by 64
0101 = Base clock value divided by 32
0100 = Base clock value divided by 16
0011 = Base clock value divided by 8
0010 = Base clock value divided by 4
0001 = Base clock value divided by 2
0000 = Base clock value

Note 1: For ROSEL (REFOCON<4>), the primary oscillator is available only when configured as the default via the FOSCx settings. This is regardless of whether the device is in Sleep mode.

6.2 PIC18 Instruction Cycle

6.2.1 CLOCKING SCHEME

The microcontroller clock input, whether from an internal or external source, is internally divided by four to generate four non-overlapping quadrature clocks (Q1, Q2, Q3 and Q4). Internally, the Program Counter is incremented on every Q1, with the instruction fetched from the program memory and latched into the Instruction Register (IR) during Q4.

The instruction is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow are shown in Figure 6-4.

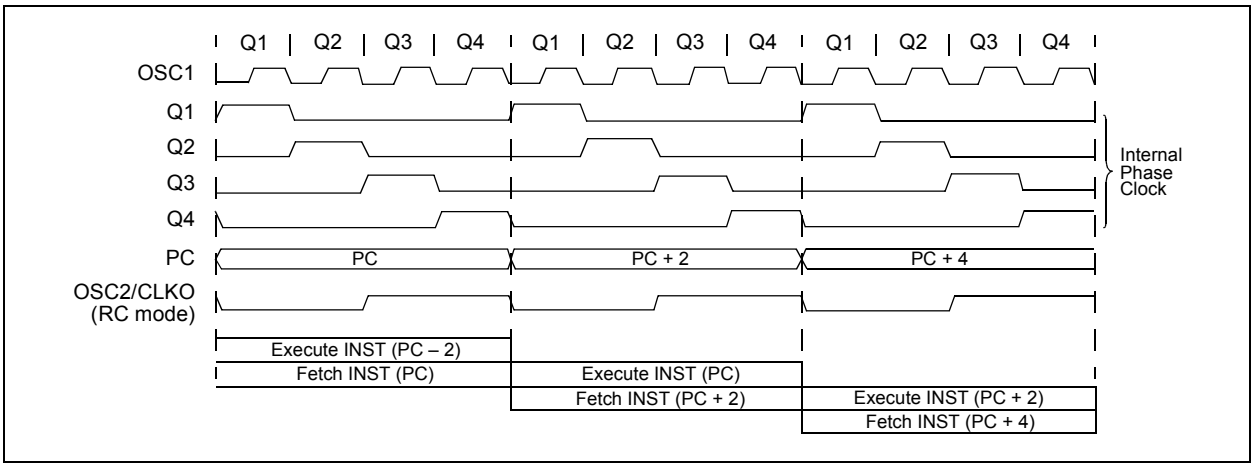
6.2.2 INSTRUCTION FLOW/PIPELINING

An “Instruction Cycle” consists of four Q cycles, Q1 through Q4. The instruction fetch and execute are pipelined in such a manner that a fetch takes one instruction cycle, while the decode and execute take another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction (such as GOTO) causes the Program Counter to change, two cycles are required to complete the instruction. (See Example 6-3.)

A fetch cycle begins with the Program Counter (PC) incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the Instruction Register (IR) in cycle, Q1. This instruction is then decoded and executed during the Q2, Q3 and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

FIGURE 6-4: CLOCK/INSTRUCTION CYCLE

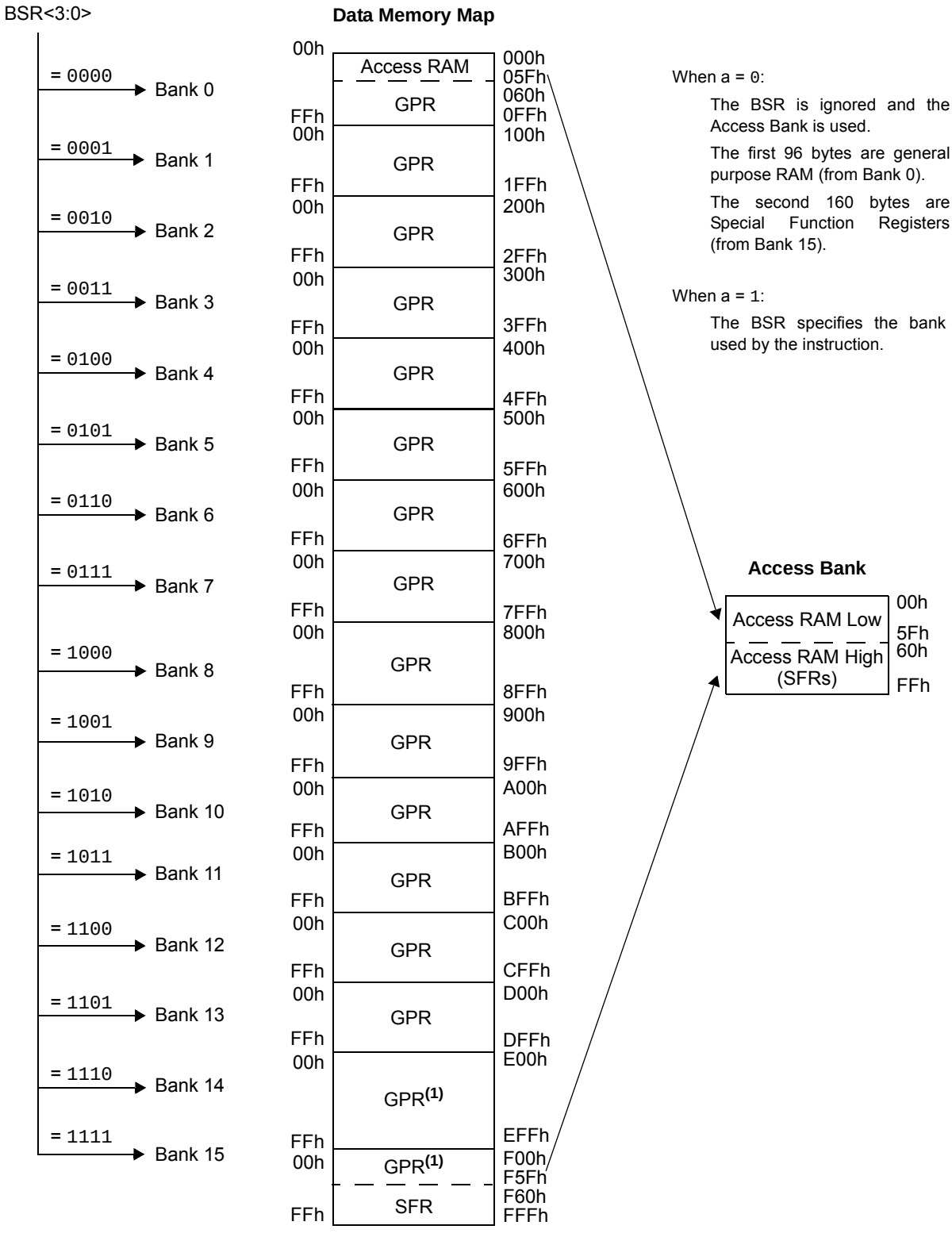


EXAMPLE 6-3: INSTRUCTION PIPELINE FLOW

	TCY0	TCY1	TCY2	TCY3	TCY4	TCY5
1. MOVLW 55h	Fetch 1	Execute 1				
2. MOVWF PORTB		Fetch 2	Execute 2			
3. BRA SUB_1			Fetch 3	Execute 3		
4. BSF PORTA, BIT3 (Forced NOP)				Fetch 4	Flush (NOP)	
5. Instruction @ address SUB_1					Fetch SUB_1	Execute SUB_1

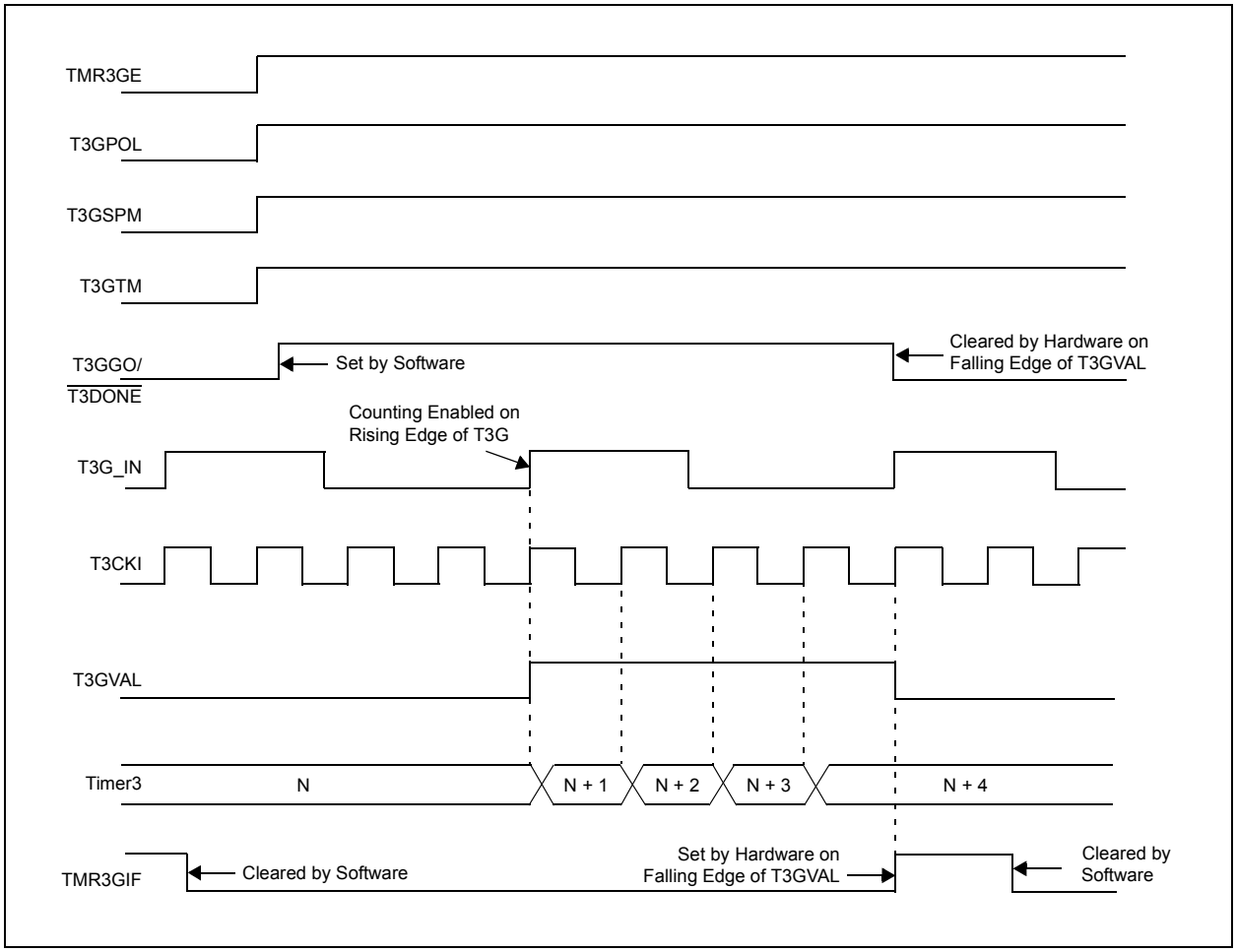
All instructions are single cycle, except for any program branches. These take two cycles since the fetch instruction is “flushed” from the pipeline while the new instruction is being fetched and then executed.

FIGURE 6-6: DATA MEMORY MAP FOR PIC18FX5K80 AND PIC18FX6K80 DEVICES



Note 1: Addresses, E41h through F5Fh, are also used by SFRs, but are not part of the Access RAM. Users must always use the complete address, or load the proper BSR value, to access these registers.

FIGURE 16-5: TIMER3 GATE SINGLE PULSE AND TOGGLE COMBINED MODE



16.5.5 TIMER3 GATE VALUE STATUS

When Timer3 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T3GVAL bit (T3GCON<2>). The T3GVAL bit is valid even when the Timer3 gate is not enabled (TMR3GE bit is cleared).

16.5.6 TIMER3 GATE EVENT INTERRUPT

When the Timer3 gate event interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T3GVAL occurs, the TMR3GIF flag bit in the PIR2 register will be set. If the TMR3GIE bit in the PIE2 register is set, then an interrupt will be recognized.

The TMR3GIF flag bit operates even when the Timer3 gate is not enabled (TMR3GE bit is cleared).

18.2 CTMU Operation

The CTMU works by using a fixed current source to charge a circuit. The type of circuit depends on the type of measurement being made.

In the case of charge measurement, the current is fixed and the amount of time the current is applied to the circuit is fixed. The amount of voltage read by the A/D becomes a measurement of the circuit's capacitance.

In the case of time measurement, the current, as well as the capacitance of the circuit, is fixed. In this case, the voltage read by the A/D is representative of the amount of time elapsed from the time the current source starts and stops charging the circuit.

If the CTMU is being used as a time delay, both capacitance and current source are fixed, as well as the voltage supplied to the comparator circuit. The delay of a signal is determined by the amount of time it takes the voltage to charge to the comparator threshold voltage.

18.2.1 THEORY OF OPERATION

The operation of the CTMU is based on the equation for charge:

$$I = C \bullet \frac{dV}{dT}$$

More simply, the amount of charge measured in coulombs in a circuit is defined as current in amperes (I) multiplied by the amount of time in seconds that the current flows (t). Charge is also defined as the capacitance in farads (C) multiplied by the voltage of the circuit (V). It follows that:

$$I \bullet t = C \bullet V$$

The CTMU module provides a constant, known current source. The A/D Converter is used to measure (V) in the equation, leaving two unknowns: capacitance (C) and time (t). The above equation can be used to calculate capacitance or time, by either the relationship using the known fixed capacitance of the circuit:

$$t = (C \bullet V)/I$$

or by:

$$C = (I \bullet t)/V$$

using a fixed time that the current source is applied to the circuit.

18.2.2 CURRENT SOURCE

At the heart of the CTMU is a precision current source, designed to provide a constant reference for measurements. The level of current is user-selectable across three ranges, or a total of two orders of magnitude, with the ability to trim the output in ±2% increments (nominal). The current range is selected by the IRNG<1:0> bits (CTMUICON<1:0>), with a value of '01' representing the lowest range.

Current trim is provided by the ITRIM<5:0> bits (CTMUICON<7:2>). These six bits allow trimming of the current source in steps of approximately 2% per step. Half of the range adjusts the current source positively and the other half reduces the current source. A value of '000000' is the neutral position (no change). A value of '100001' is the maximum negative adjustment (approximately -62%) and '011111' is the maximum positive adjustment (approximately +62%).

18.2.3 EDGE SELECTION AND CONTROL

CTMU measurements are controlled by edge events occurring on the module's two input channels. Each channel, referred to as Edge 1 and Edge 2, can be configured to receive input pulses from one of the edge input pins (CTED1 and CTED2) or CCPx Special Event Triggers (ECCP1 and CCP2). The input channels are level-sensitive, responding to the instantaneous level on the channel rather than a transition between levels. The inputs are selected using the EDG1SEL and EDG2SEL bit pairs (CTMUCONL<3:2>, 6:5>).

In addition to source, each channel can be configured for event polarity using the EDGE2POL and EDGE1POL bits (CTMUCONL<7,4>). The input channels can also be filtered for an edge event sequence (Edge 1 occurring before Edge 2) by setting the EDGSEQEN bit (CTMUCONH<2>).

18.2.4 EDGE STATUS

The CTMUCONL register also contains two status bits, EDG2STAT and EDG1STAT (CTMUCONL<1:0>). Their primary function is to show if an edge response has occurred on the corresponding channel. The CTMU automatically sets a particular bit when an edge response is detected on its channel. The level-sensitive nature of the input channels also means that the status bits become set immediately if the channel's configuration is changed and matches the channel's current state.

The module uses the edge status bits to control the current source output to external analog modules (such as the A/D Converter). Current is only supplied to external modules when only one (not both) of the status bits is set. Current is shut off when both bits are either set or cleared. This allows the CTMU to measure current only during the interval between edges. After both status bits are set, it is necessary to clear them before another measurement is taken. Both bits should be cleared simultaneously, if possible, to avoid re-enabling the CTMU current source.

In addition to being set by the CTMU hardware, the edge status bits can also be set by software. This permits a user application to manually enable or disable the current source. Setting either (but not both) of the bits enables the current source. Setting or clearing both bits at once disables the source.

19.4.2 PWM DUTY CYCLE

The PWM duty cycle is specified, to use CCP4 as an example, by writing to the CCPR4L register and to the CCP4CON<5:4> bits. Up to 10-bit resolution is available. The CCPR4L contains the eight MSbs and the CCP4CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR4L:CCP4CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

EQUATION 19-2:

$$\text{PWM Duty Cycle} = (\text{CCPR4L:CCP4CON<5:4>}) \bullet T_{OSC} \bullet (\text{TMR2 Prescale Value})$$

CCPR4L and CCP4CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR4H until after a match between PR2 and TMR2 occurs (that is, the period is complete). In PWM mode, CCPR4H is a read-only register.

The CCPR4H register and a two-bit internal latch are used to double-buffer the PWM duty cycle. This double-buffering is essential for glitchless PWM operation.

When the CCPR4H and two-bit latch match TMR2, concatenated with an internal two-bit Q clock or two bits of the TMR2 prescaler, the CCP4 pin is cleared.

The maximum PWM resolution (bits) for a given PWM frequency is given by the equation:

EQUATION 19-3:

$$\text{PWM Resolution (max)} = \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log(2)} \text{ bits}$$

Note: If the PWM duty cycle value is longer than the PWM period, the CCP4 pin will not be cleared.

TABLE 19-4: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 40 MHz

PWM Frequency	2.44 kHz	9.77 kHz	39.06 kHz	156.25 kHz	312.50 kHz	416.67 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	FFh	FFh	FFh	3Fh	1Fh	17h
Maximum Resolution (bits)	10	10	10	8	7	6.58

19.4.3 SETUP FOR PWM OPERATION

To configure the CCP module for PWM operation, using CCP4 as an example:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR4L register and CCP4CON<5:4> bits.
3. Make the CCP4 pin an output by clearing the appropriate TRIS bit.
4. Set the TMR2 prescale value, then enable Timer2 by writing to T2CON.
5. Configure the CCP4 module for PWM operation.

REGISTER 20-2: CCPTMRS: CCP TIMER SELECT REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	C5TSEL	C4TSEL	C3TSEL	C2TSEL	C1TSEL
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 7-5 **Unimplemented:** Read as '0'

bit 4 **C5TSEL:** CCP5 Timer Selection bit
0 = CCP5 is based off of TMR1/TMR2
1 = CCP5 is based off of TMR3/TMR4

bit 3 **C4TSEL:** CCP4 Timer Selection bit
0 = CCP4 is based off of TMR1/TMR2
1 = CCP4 is based off of TMR3/TMR4

bit 2 **C3TSEL:** CCP3 Timer Selection bit
0 = CCP3 is based off of TMR1/TMR2
1 = CCP3 is based off of TMR3/TMR4

bit 1 **C2TSEL:** CCP2 Timer Selection bit
0 = CCP2 is based off of TMR1/TMR2
1 = CCP2 is based off of TMR3/TMR4

bit 0 **C1TSEL:** CCP1 Timer Selection bit
0 = ECCP1 is based off of TMR1/TMR2
1 = ECCP1 is based off of TMR3/TMR4

TABLE 22-4: BAUD RATES FOR ASYNCHRONOUS MODES

BAUD RATE (K)	SYNC = 0, BRGH = 0, BRG16 = 0											
	Fosc = 64.000 MHz			Fosc = 40.000 MHz			Fosc = 20.000 MHz			Fosc = 10.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	—	—	—	—	—	—
1.2	—	—	—	—	—	—	1.221	1.73	255	1.202	0.16	129
2.4	—	—	—	2.441	1.73	255	2.404	0.16	129	2.404	0.16	64
9.6	9.615	0.16	103	9.615	0.16	64	9.766	1.73	31	9.766	1.73	15
19.2	19.231	0.16	51	19.531	1.73	31	19.531	1.73	15	19.531	1.73	7
57.6	58.824	2.13	16	56.818	-1.36	10	62.500	8.51	4	52.083	-9.58	2
115.2	111.111	-3.55	8	125.000	8.51	4	104.167	-9.58	2	78.125	-32.18	1

BAUD RATE (K)	SYNC = 0, BRGH = 0, BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 2.000 MHz			Fosc = 1.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	0.300	0.16	207	0.300	-0.16	103	0.300	-0.16	51
1.2	1.201	-0.16	103	1.202	0.16	51	1.201	-0.16	25	1.201	-0.16	12
2.4	2.403	-0.16	51	2.404	0.16	25	2.403	-0.16	12	—	—	—
9.6	9.615	-0.16	12	8.929	-6.99	6	—	—	—	—	—	—
19.2	—	—	—	20.833	8.51	2	—	—	—	—	—	—
57.6	—	—	—	62.500	8.51	0	—	—	—	—	—	—
115.2	—	—	—	62.500	-45.75	0	—	—	—	—	—	—

BAUD RATE (K)	SYNC = 0, BRGH = 1, BRG16 = 0											
	Fosc = 64.000 MHz			Fosc = 40.000 MHz			Fosc = 20.000 MHz			Fosc = 10.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	—	—	—	—	—	—
1.2	—	—	—	—	—	—	—	—	—	—	—	—
2.4	—	—	—	—	—	—	—	—	—	2.441	1.73	255
9.6	—	—	—	9.766	1.73	255	9.615	0.16	129	9.615	0.16	64
19.2	19.417	1.13	207	19.231	0.16	129	19.231	0.16	64	19.531	1.73	31
57.6	59.701	3.65	68	58.140	0.94	42	56.818	-1.36	21	56.818	-1.36	10
115.2	121.212	5.22	34	113.636	-1.36	21	113.636	-1.36	10	125.000	8.51	4

BAUD RATE (K)	SYNC = 0, BRGH = 1, BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 2.000 MHz			Fosc = 1.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	—	—	—	0.300	-0.16	207
1.2	—	—	—	1.202	0.16	207	1.201	-0.16	103	1.201	-0.16	51
2.4	2.403	-0.16	207	2.404	0.16	103	2.403	-0.16	51	2.403	-0.16	25
9.6	9.615	-0.16	51	9.615	0.16	25	9.615	-0.16	12	—	—	—
19.2	19.230	-0.16	25	19.231	0.16	12	—	—	—	—	—	—
57.6	55.555	3.55	8	62.500	8.51	3	—	—	—	—	—	—
115.2	—	—	—	125.000	8.51	1	—	—	—	—	—	—

TABLE 22-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR1	PSPIF	ADIF	RC1IF	TX1IF	SSPIF	TMR1GIF	TMR2IF	TMR1IF
PIE1	PSPIE	ADIE	RC1IE	TX1IE	SSPIE	TMR1GIE	TMR2IE	TMR1IE
IPR1	PSPIP	ADIP	RC1IP	TX1IP	SSPIP	TMR1GIP	TMR2IP	TMR1IP
PIR3	—	—	RC2IF	TX2IF	CTMUIF	CCP2IF	CCP1IF	—
PIE3	—	—	RC2IE	TX2IE	CTMUIE	CCP2IE	CCP1IE	—
IPR3	—	—	RC2IP	TX2IP	CTMUIP	CCP2IP	CCP1IP	—
RCSTA1	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
RCREG1	EUSART1 Receive Register							
TXSTA1	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN
SPBRGH1	EUSART1 Baud Rate Generator Register High Byte							
SPBRG1	EUSART1 Baud Rate Generator Register Low Byte							
RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
RCREG2	EUSART2 Receive Register							
TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN
SPBRGH2	EUSART2 Baud Rate Generator Register High Byte							
SPBRG2	EUSART2 Baud Rate Generator Register Low Byte							
PMD0	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	UART2MD	UART1MD	SSPMD
ODCON	SSPOD	CCP5OD	CCP4OD	CCP3OD	CCP2OD	CCP1OD	U2OD	U1OD

Legend: — = unimplemented, read as '0'. Shaded cells are not used for synchronous master reception.

22.5 EUSARTx Synchronous Slave Mode

Synchronous Slave mode is entered by clearing bit, CSRC (TXSTAx<7>). This mode differs from the Synchronous Master mode in that the shift clock is supplied externally at the CKx pin (instead of being supplied internally in Master mode). This allows the device to transfer or receive data while in any low-power mode.

22.5.1 EUSARTx SYNCHRONOUS SLAVE TRANSMISSION

The operation of the Synchronous Master and Slave modes is identical, except in the case of Sleep mode.

If two words are written to the TXREGx and then the SLEEP instruction is executed, the following will occur:

a) The first word will immediately transfer to the TSR register and transmit.
b) The second word will remain in the TXREGx register.
c) Flag bit, TXxIF, will not be set.
d) When the first word has been shifted out of TSR, the TXREGx register will transfer the second word to the TSR and flag bit, TXxIF, will now be set.
e) If enable bit, TXxIE, is set, the interrupt will wake the chip from Sleep. If the global interrupt is enabled, the program will branch to the interrupt vector.

To set up a Synchronous Slave Transmission:

1. Enable the synchronous slave serial port by setting bits, SYNC and SPEN, and clearing bit, CSRC.
2. Clear bits, CREN and SREN.
3. If interrupts are desired, set enable bit, TXxIE.
4. If 9-bit transmission is desired, set bit, TX9.
5. Enable the transmission by setting enable bit, TXEN.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit, TX9D.
7. Start transmission by loading data to the TXREGx register.
8. If using interrupts, ensure that the GIE and PEIE bits (INTCON<7:6>) are set.

TABLE 22-10: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR1	PSPIF	ADIF	RC1IF	TX1IF	SSPIF	TMR1GIF	TMR2IF	TMR1IF
PIE1	PSPIE	ADIE	RC1IE	TX1IE	SSPIE	TMR1GIE	TMR2IE	TMR1IE
IPR1	PSPIP	ADIP	RC1IP	TX1IP	SSPIP	TMR1GIP	TMR2IP	TMR1IP
PIR3	—	—	RC2IF	TX2IF	CTMUIF	CCP2IF	CCP1IF	—
PIE3	—	—	RC2IE	TX2IE	CTMUIE	CCP2IE	CCP1IE	—
IPR3	—	—	RC2IP	TX2IP	CTMUIP	CCP2IP	CCP1IP	—
RCSTA1	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
TXREG1	EUSART1 Transmit Register							
TXSTA1	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN
SPBRGH1	EUSART1 Baud Rate Generator Register High Byte							
SPBRG1	EUSART1 Baud Rate Generator Register Low Byte							
RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
TXREG2	EUSART2 Transmit Register							
TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN
SPBRGH2	EUSART2 Baud Rate Generator Register High Byte							
SPBRG2	EUSART2 Baud Rate Generator Register Low Byte							
PMD0	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	UART2MD	UART1MD	SSPMD
ODCON	SSPOD	CCP5OD	CCP4OD	CCP3OD	CCP2OD	CCP1OD	U2OD	U1OD

Legend: — = unimplemented, read as '0'. Shaded cells are not used for synchronous slave transmission.

23.4 Selecting and Configuring Automatic Acquisition Time

The ADCON2 register allows the user to select an acquisition time that occurs each time the GO/DONE bit is set.

When the GO/DONE bit is set, sampling is stopped and a conversion begins. The user is responsible for ensuring the required acquisition time has passed between selecting the desired input channel and setting the GO/DONE bit.

This occurs when the ACQT<2:0> bits (ADCON2<5:3>) remain in their Reset state ('000'), which is compatible with devices that do not offer programmable acquisition times.

If desired, the ACQTx bits can be set to select a programmable acquisition time for the A/D module. When the GO/DONE bit is set, the A/D module continues to sample the input for the selected acquisition time, then automatically begins a conversion. Since the acquisition time is programmed, there may be no need to wait for an acquisition time between selecting a channel and setting the GO/DONE bit.

In either case, when the conversion is completed, the GO/DONE bit is cleared, the ADIF flag is set and the A/D begins sampling the currently selected channel again. If an acquisition time is programmed, there is nothing to indicate if the acquisition time has ended or if the conversion has begun.

23.5 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires 14 TAD per 12-bit conversion. The source of the A/D conversion clock is software selectable.

The possible options for TAD are:

- 2 TOSC
- 4 TOSC
- 8 TOSC
- 16 TOSC
- 32 TOSC
- 64 TOSC
- Using the internal RC Oscillator

For correct A/D conversions, the A/D conversion clock (TAD) must be as short as possible but greater than the minimum TAD. (For more information, see Parameter 130 in Table 31-26.)

Table 23-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

TABLE 23-1: TAD vs. DEVICE OPERATING FREQUENCIES

AD Clock Source (TAD)		Maximum Device Frequency
Operation	ADCS<2:0>	
2 TOSC	000	2.50 MHz
4 TOSC	100	5.00 MHz
8 TOSC	001	10.00 MHz
16 TOSC	101	20.00 MHz
32 TOSC	010	40.00 MHz
64 TOSC	110	64.00 MHz
RC(2)	x11	1.00 MHz(1)

Note 1: The RC source has a typical TAD time of 4 μs.

2: For device frequencies above 1 MHz, the device must be in Sleep mode for the entire conversion or the A/D accuracy may be out of specification.

23.6 Configuring Analog Port Pins

The ANCON0, ANCON1, TRISA, TRISB, TRISC and TRISC registers control the operation of the A/D port pins. The port pins needed as analog inputs must have their corresponding TRISx bits set (input). If the TRISx bit is cleared (output), the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the CHS<3:0> bits and the TRISx bits.

Note:	When reading the PORT register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs will convert an analog input. Analog levels on a digitally configured input will be accurately converted. Analog levels on any pin defined as a digital input may cause the digital input buffer to consume current out of the device's specification limits.

REGISTER 27-14: RXB1CON: RECEIVE BUFFER 1 CONTROL REGISTER

Mode 0	R/C-0	R/W-0	R/W-0	U-0	R-0	R/W-0	R-0	R-0
	RXFUL(1)	RXM1	RXM0	—	RXRTRRO	FILHIT2	FILHIT1	FILHIT0

Mode 1,2	R/C-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXFUL(1)	RXM1	RTRRO	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0
	bit 7							bit 0

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

bit 7 **RXFUL:** Receive Full Status bit(1)
1 = Receive buffer contains a received message
0 = Receive buffer is open to receive a new message

bit 6-5, 6 Mode 0:
RXM<1:0>: Receive Buffer Mode bit 1 (combines with RXM0 to form RXM<1:0> bits, see bit 5)
11 = Receive all messages (including those with errors); filter criteria is ignored
10 = Receive only valid messages with extended identifier; EXIDEN in RXFnSIDL must be '1'
01 = Receive only valid messages with standard identifier, EXIDEN in RXFnSIDL must be '0'
00 = Receive all valid messages as per EXIDEN bit in RXFnSIDL register
Mode 1, 2:
RXM1: Receive Buffer Mode bit
1 = Receive all messages (including those with errors); acceptance filters are ignored
0 = Receive all valid messages as per acceptance filters

bit 5 Mode 0:
RXM<1:0>: Receive Buffer Mode bit 0 (combines with RXM1 to form RXM<1:0> bits, see bit 6)
Mode 1, 2:
RTRRO: Remote Transmission Request bit for Received Message (read-only)
1 = A remote transmission request is received
0 = A remote transmission request is not received

bit 4 Mode 0:
FILHIT2**4:** Filter Hit bit 4
Mode 1, 2:
FILHIT<4:0>: Filter Hit bit 4
This bit combines with other bits to form the filter acceptance bits<4:0>.

bit 3 Mode 0:
RXRTRRO: Remote Transmission Request bit for Received Message (read-only)
1 = A remote transmission request is received
0 = A remote transmission request is not received
Mode 1, 2:
FILHIT<4:0>: Filter Hit bit 3
This bit combines with other bits to form the filter acceptance bits<4:0>.

Note 1: This bit is set by the CAN module upon receiving a message and must be cleared by software after the buffer is read. As long as RXFUL is set, no new message will be loaded and the buffer will be considered full.

REGISTER 27-51: MSEL3: MASK SELECT REGISTER 3(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
FIL15_1	FIL15_0	FIL14_1	FIL14_0	FIL13_1	FIL13_0	FIL12_1	FIL12_0
bit 7							bit 0

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

bit 7-6 **FIL15_<1:0>:** Filter 15 Select bits 1 and 0
- 11 = No mask
- 10 = Filter 15
- 01 = Acceptance Mask 1
- 00 = Acceptance Mask 0

bit 5-4 **FIL14_<1:0>:** Filter 14 Select bits 1 and 0
- 11 = No mask
- 10 = Filter 15
- 01 = Acceptance Mask 1
- 00 = Acceptance Mask 0

bit 3-2 **FIL13_<1:0>:** Filter 13 Select bits 1 and 0
- 11 = No mask
- 10 = Filter 15
- 01 = Acceptance Mask 1
- 00 = Acceptance Mask 0

bit 1-0 **FIL12_<1:0>:** Filter 12 Select bits 1 and 0
- 11 = No mask
- 10 = Filter 15
- 01 = Acceptance Mask 1
- 00 = Acceptance Mask 0

Note 1: This register is available in Mode 1 and 2 only.

REGISTER 28-2: CONFIG1H: CONFIGURATION REGISTER 1 HIGH (BYTE ADDRESS 300001h)

R/P-0	R/P-0	U-0	U-0	R/P-1	R/P-0	R/P-0	R/P-0
IESO	FCMEN	—	PLLCFG(1)	FOSC3(2)	FOSC2(2)	FOSC1(2)	FOSC0(2)
bit 7							bit 0

Legend:	P = Programmable 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 **IESO:** Internal/External Oscillator Switchover bit
1 = Two-Speed Start-up is enabled
0 = Two-Speed Start-up is disabled

bit 6 **FCMEN:** Fail-Safe Clock Monitor Enable bit
1 = Fail-Safe Clock Monitor is enabled
0 = Fail-Safe Clock Monitor is disabled

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

bit 4 **PLLCFG:** 4X PLL Enable bit(1)
1 = Oscillator is multiplied by 4
0 = Oscillator is used directly

bit 3-0 **FOSC<3:0>:** Oscillator Selection bits(2)
1101 = EC1, EC oscillator **(low power, DC-160 kHz)**
1100 = EC1IO, EC oscillator with CLKOUT function on RA6 **(low power, DC-160 kHz)**
1011 = EC2, EC oscillator **(medium power, 160 kHz-16 MHz)**
1010 = EC2IO, EC oscillator with CLKOUT function on RA6 **(medium power, 160 kHz-16 MHz)**
0101 = EC3, EC oscillator **(high power, 16 MHz-64 MHz)**
0100 = EC3IO, EC oscillator with CLKOUT function on RA6 **(high power, 16 MHz-64 MHz)**
0011 = HS1, HS oscillator **(medium power, 4 MHz-16 MHz)**
0010 = HS2, HS oscillator **(high power, 16 MHz-25 MHz)**
0001 = XT oscillator
0000 = LP oscillator
0111 = RC, external RC oscillator
0110 = RCIO, external RC oscillator with CKLOUT function on RA6
1000 = INTIO2, internal RC oscillator
1001 = INTIO1, internal RC oscillator with CLKOUT function on RA6

Note 1: Not valid for the INTIOx PLL mode.
2: INTIO + PLL can be enabled only by the PLLEN bit (OSCTUNE<6>). Other PLL modes can be enabled by either the PLLEN bit or the PLLCFG (CONFIG1H<4>) bit.

MOVFF **Move f to f**

Syntax: MOVFF f_s,f_d

Operands: $0 \le f_s \le 4095$
$0 \le f_d \le 4095$

Operation: $(f_s) \rightarrow f_d$

Status Affected: None

Encoding:

1st word (source)	1100	ffff	ffff	$ffff_s$
2nd word (destin.)	1111	ffff	ffff	$ffff_d$

Description: The contents of source register 'f_s' are moved to destination register 'f_d'. Location of source 'f_s' can be anywhere in the 4096-byte data space (000h to FFFh) and location of destination 'f_d' can also be anywhere from 000h to FFFh.

Either source or destination can be W (a useful special situation).

MOVFF is particularly useful for transferring a data memory location to a peripheral register (such as the transmit buffer or an I/O port).

The MOVFF instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f' (src)	Process Data	No operation
Decode	No operation No dummy read	No operation	Write register 'f' (dest)

Example:
```
MOVFF   REG1, REG2
```
Before Instruction
REG1 = 33h
REG2 = 11h
After Instruction
REG1 = 33h
REG2 = 33h

MOVLB **Move Literal to Low Nibble in BSR**

Syntax: MOVLW k

Operands: $0 \le k \le 255$

Operation: $k \rightarrow BSR$

Status Affected: None

Encoding:

0000	0001	kkkk	kkkk

Description: The eight-bit literal 'k' is loaded into the Bank Select Register (BSR). The value of BSR<7:4> always remains '0' regardless of the value of k_7:k_4.

Words: 1

Cycles: 1

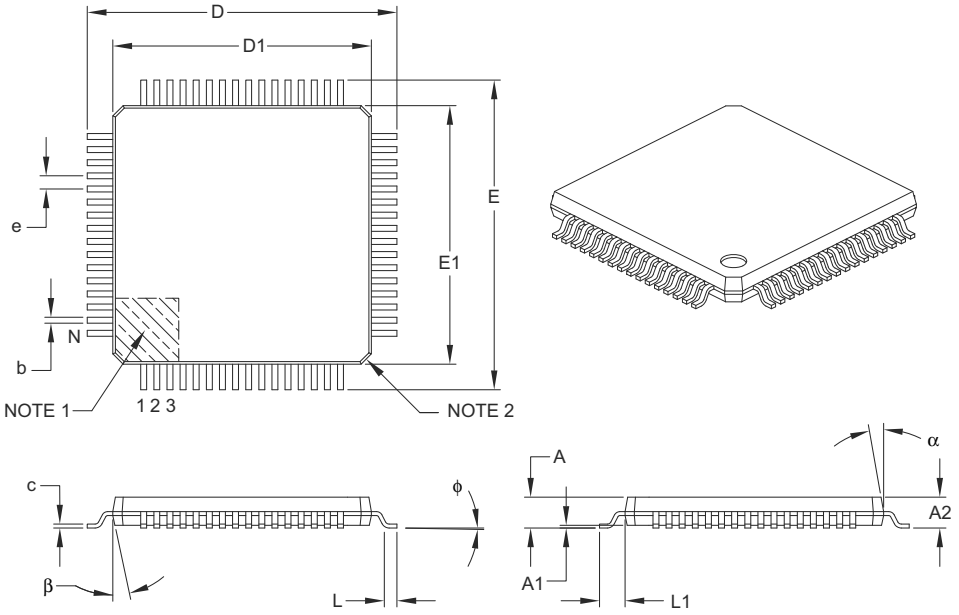
Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write literal 'k' to BSR

Example:
```
MOVLB   5
```
Before Instruction
BSR Register = 02h
After Instruction
BSR Register = 05h

64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		64	
Lead Pitch	e		0.50 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E		12.00 BSC	
Overall Length	D		12.00 BSC	
Molded Package Width	E1		10.00 BSC	
Molded Package Length	D1		10.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions 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.

Microchip Technology Drawing C04-085B