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	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	13
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 8x8b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c781t-i-so

Peripheral Features (Continued):

- Dual Analog Comparator module with:
  - Individual enable and interrupt bits
  - Programmable speed and output polarity
  - Fully configurable inputs and outputs
  - Reference from DAC, or VREF1/VREF2
  - Low input offset voltage.
- VR voltage reference module:
  - 3.072V +/- 0.7% @25°C, AVDD = 5V
  - Configurable output to ADC reference, DAC reference, and VR pin
  - 5 mA sink/source
- Programmable Switch Mode Controller module:
  - PWM and PSM modes
  - Programmable switching frequency
  - Configurable for either single or dual feedback inputs
  - Configurable single or dual outputs
  - Slope compensation output available in single output mode

Key Features PIC® Mid-Range Reference Manual (DS33023)	PIC16C781	PIC16C782
Operating Frequency	DC - 20 MHz	DC - 20 MHZ
RESETS (and Delays)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)
Program Memory (14 bit words)	1K	2K
Data Memory (bytes)	128	128
Interrupts	8	8
I/O Ports	13 + 3 Input only	13 + 3 Input only
Timers	2	2
Programmable Switch Mode Controller	1	1
8-bit Analog-to-Digital Module	1	1
ADC channels	8 External, 2 Internal	8 External, 2 Internal
8-bit Digital-to-Analog Module	1	1
Comparators	2	2
Comparator Channels	4 (AN<7:4>)	4 (AN<7:4>)
Operational Amplifier	1	1
Voltage Reference	1	1
Brown-out Reset	Yes	Yes
Programmable Low Voltage Detect	Yes	Yes
Instruction Set	35 Instructions	35 Instructions

TABLE 1-1: PIC16C781/782 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/OPA+	RA0	ST	N/A	Port Input
	AN0	AN	—	ADC Input
	OPA+	AN	—	OPAMP Non-inverting Input
RA1/AN1/OPA-	RA1	ST	N/A	Port Input
	AN1	AN	—	ADC Input
	OPA-	AN	—	OPAMP Inverting Input
RA2/AN2/VREF2	RA2	ST	CMOS	Bi-directional I/O
	AN2	AN	—	ADC Input
	VREF2	AN	—	Comparator 2 Voltage Reference Input
RA3/AN3/VREF1	RA3	ST	CMOS	Bi-directional I/O
	AN3	AN	—	ADC Input
	VREF1	AN	—	Comparator 1, ADC, DACREF Input
RA4/T0CKI	RA4	ST	OD	Bi-directional I/O
	TOCKI	ST	—	Timer0 Clock Input
RA5/$\overline{\text{MCLR}}$/VPP	RA5	ST	N/A	Port Input
	$\overline{\text{MCLR}}$	ST	—	Master Clear Input
	VPP	Power	—	Programming Voltage
RA6/OSC2/CLKOUT/T1CKI	RA6	ST	CMOS	Bi-directional I/O
	OSC2	—	XTAL	Crystal/Resonator
	CLKOUT	—	CMOS	Fosc/4 Output
	T1CKI	ST	—	Timer1 Clock Input
RA7/OSC1/CLKIN	RA7	ST	CMOS	Bi-directional I/O
	OSC1	XTAL	—	Crystal/Resonator
	CLKIN	ST	—	External Clock Input
RB0/INT/AN4/VR	RB0	TTL	CMOS	Bi-directional I/O
	INT	ST	—	External Interrupt
	AN4	AN	—	ADC, Comparator Input
	VR	—	AN	Internal Voltage Reference Output
RB1/AN5/VDAC	RB1	TTL	CMOS	Bi-directional I/O
	AN5	AN	—	ADC, Comparator Input
	VDAC	—	AN	DAC Output
RB2/AN6	RB2	TTL	CMOS	Bi-directional I/O
	AN6	AN	—	ADC, Comparator Input
RB3/AN7/OPA	RB3	TTL	CMOS	Bi-directional I/O
	AN7	AN	—	ADC, Comparator Input
	OPA	—	AN	OPAMP Output
RB4	RB4	TTL	CMOS	Bi-directional I/O
RB5	RB5	TTL	CMOS	Bi-directional I/O
RB6/C1/PSMC1A	RB6	TTL	CMOS	Bi-directional I/O
	C1	—	CMOS	Comparator 1 Output
	PSMC1A	—	CMOS	PSMC Output 1A

TABLE 2-1: PIC16C781/782 SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on Page:
Bank 2											
100h(2)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	23
101h	TMR0	Timer0 Module's Register								xxxx xxxx	51
102h(2)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	23
103h(2)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	17
104h(2)	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	23
105h	—	Unimplemented								—	—
106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx 0000	35
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	23
10Bh(2)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	19
10Ch	PMDATL	PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0	0000 0000	48
10Dh	PMADRL	PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0	xxxx xxxx	48
10Eh	PMDATH	—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8	--00 0000	47
10Fh	PMADRH	—	—	—	Reserved	Reserved	PMA10	PMA9	PMA8	---x xxxx	48
110h	CALCON	CAL	CALERR	CALREF	—	—	—	—	—	000- ----	85
111h	PSMCCON0	SMCCL1	SMCCL0	MINDC1	MINDC0	MAXDC1	MAXDC0	DC1	DC0	0000 0000	104
112h	PSMCCON1	SMCON	S1APOL	S1BPOL	—	SCEN	SMCOM	PWM/$\overline{PSM}$	SMCCS	000- 0000	104
113h	—	Unimplemented								—	—
114h	—	Unimplemented								—	—
115h	—	Unimplemented								—	—
116h	—	Unimplemented								—	—
117h	—	Unimplemented								—	—
118h	—	Unimplemented								—	—
119h	CM1CON0	C1ON	C1OUT	C1OE	C1POL	C1SP	C1R	C1CH1	C1CH0	0000 0000	91
11Ah	CM2CON0	C2ON	C2OUT	C2OE	C2POL	C2SP	C2R	C2CH1	C2CH0	0000 0000	93
11Bh	CM2CON1	MC1OUT	MC2OUT	—	—	—	—	—	C2SYNC	00-- ---0	94
11Ch	OPACON	OPAON	CMPEN	—	—	—	—	—	GBWP	00-- ---0	84
11Dh	—	Unimplemented								—	—
11Eh	DAC	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	0000 0000	79
11Fh	DACON0	DAON	DAOE	—	—	—	—	DARS1	DARS0	00-- --00	79

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter. See Section 2.9 for more detail.

2: These registers can be addressed from any bank.

2.5 INTCON Register

The INTCON register is a readable and writable register which contains:

- Enable and interrupt flag bits for TMR0 register overflow
- Enable and interrupt flag bits for the external interrupt (INT)
- Enable and interrupt flag bits for PORTB Interrupt-on-Change (IOCB)
- Peripheral interrupt enable bit
- Global interrupt enable bit

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTERRUPT CONTROL REGISTER (INTCON: 0Bh, 8Bh, 10Bh, 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T01E	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3 **RBIE:** RB Port Change Interrupt Enable bit(1)
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0 **RBIF:** RB Port Change Interrupt Flag bit(1)
1 = When at least one of the RB7:RB0 pins changed state (must be cleared in software)
0 = None of the RB7:RB0 pins have changed state

Note 1: Individual RB pin interrupt-on-change can be enabled/disabled from the Interrupt-on-Change PORTB register (IOCB).

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

REGISTER 3-2: WEAK PULL-UP PORTB REGISTER (WPUB: 95h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
bit7							bit0

bit 7-0 **WPUB<7:0>:** PORTB Weak Pull-Up Control bits

1 = Weak pull-up enabled for corresponding pin

0 = Weak pull-up disabled for corresponding pin

Note 1: For the WPUB register setting to take effect, the $\overline{\text{RBPU}}$ bit in the OPTION_REG register must be cleared.

2: The weak pull-up device is automatically disabled if the pin is in output mode, i.e., (TRISB = 0) for corresponding pin.

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

REGISTER 3-3: INTERRUPT-ON-CHANGE PORTB REGISTER (IOCB: 96h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0
bit7							bit0

bit 7-0 **IOCB<7:0>:** Interrupt-on-Change PORTB Control bits

1 = Interrupt-on-change enabled for corresponding pin

0 = Interrupt-on-change disabled for corresponding pin

Note 1: The interrupt enable bits, GIE and RBIE in the INTCON register, must be set for individual interrupts to be recognized.

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

3.3.3 TRISB, ANSEL, AND CONTROL PRECEDENCE

The ANSEL and TRISB registers are the primary controls for the configuration of PORTB pins. TRISB tri-states the output drivers of PORTB, and the ANSEL register disables the input buffers. It is important to program both registers when configuring a port pin, since most peripherals do not have precedence over the TRISB and ANSEL registers' control of the pin. Even if a peripheral has the ability to override the control of the TRISB and ANSEL registers, it is good practice to program both registers appropriately.

Note 1: Upon RESET, the ANSEL register configures the RB<3:0> pins as analog inputs.

2: When programmed as analog inputs, RB<3:0> pins will read as '0'.

3: There are specific cases in which the functions of the TRISB and ANSEL registers can be overridden by a peripheral or configuration word (see Figure 3-9 through Figure 3-16 for details).

4.0 PROGRAM MEMORY READ (PMR)

Program memory is readable during normal operation (full VDD range). It is read by indirect addressing through the following Special Function Registers:

- PMCON1: Control
- PMDATH: Data High
- PMDATL: Data Low
- PMADRH: Address High
- PMADRL: Address Low

When interfacing to the program memory block, the PMDATH and PMDATL registers form a 2-byte word, which holds the 14-bit data. The PMADRH and PMADRL registers form a 2-byte word, which holds the 12-bit address of the program memory location being accessed. Mid-range devices have up to 8K words of program EPROM with an address range from 0h to 3FFFh. When the device contains less memory than the full address range of the PMADRH:PMARDL registers, the Most Significant bits of the PMADRH register are ignored.

4.1 PMCON1 Register

PMCON1 is the control register for program memory accesses.

Control bit RD initiates a read operation. This bit cannot be cleared, only set, in software. It is cleared in hardware at completion of the read operation.

4.2 PMDATH and PMDATL Registers

The PMDATH:PMDATL registers are loaded with the contents of program memory addressed by the PMADRH and PMADRL registers upon completion of a Program Memory Read command.

REGISTER 4-1: PROGRAM MEMORY READ CONTROL REGISTER 1 (PMCON1: 18Ch)

R-1	U-0	U-0	U-0	U-0	U-0	U-0	R/S-0
Reserved	—	—	—	—	—	—	RD
bit7							bit0

bit 7 **Reserved:** Read as ‘1’

bit 6-1 **Unimplemented:** Read as '0

bit 0 **RD**: Read Control bit

1 = Initiates a Program memory read (read takes 2 cycles, RD is cleared in hardware)
0 = Reserved

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

REGISTER 4-2: PROGRAM MEMORY DATA HIGH (PMDATH: 10Eh)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
bit7							bit0

bit 7-6 **Unimplemented:** Read as '0

bit 5-0 **PMD<13:8>:** Program Memory Data bits
The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

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

4.4 Program Memory Read With Code Protect Set

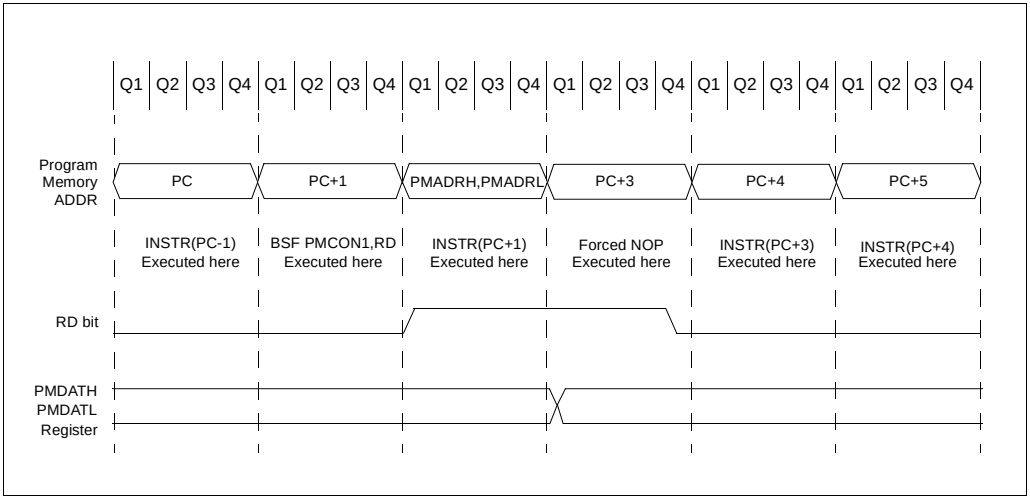
When the device is code protected, the CPU can still perform the program memory read function.

TABLE 4-1: SUMMARY OF REGISTERS ASSOCIATED WITH PMR

Address	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
10Ch	PMDATL	PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0	0000 0000	0000 0000
10Dh	PMADRL	PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0	xxxx xxxx	uuuu uuuu
10Eh	PMDATH	—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8	--00 0000	--00 0000
10Fh	PMADRH	—	—	—	Reserved	Reserved	PMA10	PMA9	PMA8	---x xxxx	---u uuuu
18Ch	PMCON1	Reserved	—	—	—	—	—	—	RD	1--- ---0	1--- ---0

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by PMR.

FIGURE 4-1: PROGRAM MEMORY READ CYCLE EXECUTION



5.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- Internal or external clock select
- Edge select for external clock
- 8-bit software programmable prescaler
- Interrupt on overflow from FFh to 00h

Figure 5-1 is a simplified block diagram of the Timer0 module.

Additional information on timer modules is available in the PIC Mid-Range Reference Manual, (DS33023).

5.1 Timer0 Operation

Timer0 can operate as either a timer or a counter.

Programming Timer0 is via the OPTION register (see Register 2-2).

Timer0 mode is selected by clearing/setting the bit T0CS (OPTION_REG<5>). In Timer mode (T0CS = 0), the Timer0 module increments every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In Counter mode, Timer0 increments either on every rising, or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge, setting selects the falling edge. Restrictions on the external clock input are discussed below.

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal system clock. Also, there is a delay in the actual incrementing of Timer0 after synchronization.

Additional information on external clock requirements is available in the PIC Mid-Range Reference Manual, (DS33023).

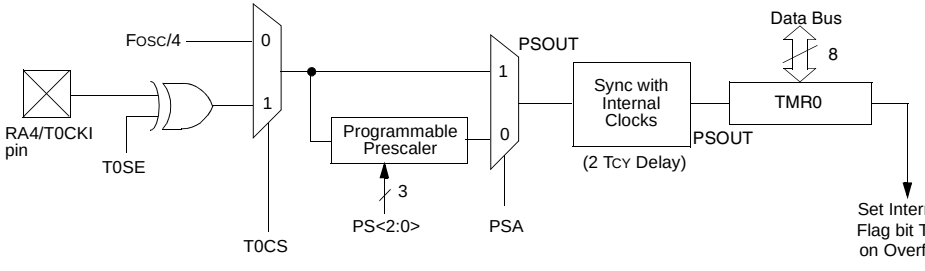
EXAMPLE 5-1: INITIALIZING TIMER0

```
;* This code block will configure Timer0
;* for Polling, internal clock & 1:16
;*  prescaler
;*
;* Wait for TMR0 overflow code included
    BANKSEL     TMR0            ; Select Bank 0
    CLRF        TMR0            ; Clear Timer0
                                ; Register

    BANKSEL     OPTION_REG      ; Select Bank 1

    MOVLW       B'11000011'     ; INT on L2H
    MOVWF       OPTION_REG      ; Internal clk,
                                ; pscaler 1:16
*****************************************
;*  Wait for TMR0 overflow
;*
T0_OVFL_WAIT
    TBFSS       INTCON,T0IF     ; Check for TMR0
                                ; overflow
    GOTO        T0_OVFL_WAIT    ; If clear, test
                                ; again
    BCF         INTCON,T0IF     ; Clear interrupt
```

FIGURE 5-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS<2:0> (OPTION_REG<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 5-2 for detailed block diagram).

NOTES:

NOTES:

9.1 Control Registers

The ADC module has three registers. These registers are:

- ADC Result Register: ADRES
- ADC Control Register 0: ADCON0
- ADC Control Register 1: ADCON1

The ADCON0 register, shown in Register 9-1, controls the operations and input channel selection for the ADC module. The ADCON1 register, shown in Register 9-3, selects the voltage reference used by the ADC module. The ADRES register, shown in Register 9-2, holds the 8-bit result of the conversion.

Additional information on using the ADC module can be found in the PIC Mid-Range MCU Family Reference Manual (DS33023) and in Application Note AN546 (DS00546).

9.1.1 ADCON0 REGISTER

The ADCON0 register, shown in Register 9-1, controls the following:

- Clock source and prescaler
- Input channel
- Conversion start/stop
- Enabling of the ADC module

Setting the ADON bit, ADCON0<0>, enables the ADC module. Clearing ADON disables the module and terminates any conversion in process.

The ADCS<1:0> bits (ADCON0<7:6>) determine the clock source used by the ADC module.

The CHS<3:0> bits (ADCON0<5:3,1>) determine the input channel to the ADC module. CHS<3> specifically determines whether the source is internal or external.

Setting the GO/$\overline{\text{DONE}}$ bit (ADCON0<2>) initiates the conversion process. The ADC clears this bit at the completion of the conversion process.

REGISTER 9-1: ADC CONTROL REGISTER 0 (ADCON0: 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/S-0	R/W-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	CHS3	ADON
bit 7							bit 0

bit 7-6 **ADCS<1:0>:** ADC Conversion Clock Select bits
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = ADRC (clock derived from a dedicated RC oscillator)

bit 5-3 **CHS<2:0>:** Analog Channel Select bits (select which channel to convert)

If CHS3 = 0:	If CHS3 = 1:
000 = channel 0 (AN0)	000 = VR
001 = channel 1 (AN1)	001 = VDAC
010 = channel 2 (AN2)	010 = Reserved. Do not use.
011 = channel 3 (AN3)	011 = Reserved. Do not use.
100 = channel 4 (AN4)	100 = Reserved. Do not use.
101 = channel 5 (AN5)	101 = Reserved. Do not use.
110 = channel 6 (AN6)	110 = Reserved. Do not use.
111 = channel 7 (AN7)	111 = Reserved. Do not use.

bit 2 **GO/$\overline{\text{DONE}}$:** ADC Conversion Status bit
1 = ADC conversion cycle in progress. Setting this bit starts an ADC conversion cycle.
0 = ADC conversion is not in progress (this bit is cleared by hardware when conversion is complete)

bit 1 **CHS3:** Analog Channel Select bit
1 = Internal channel selected for conversion
0 = External channel selected for conversion

bit 0 **ADON:** ADC On bit
1 = ADC enabled
0 = ADC disabled

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

10.3 DAC Configuration

Example 10-1 shows a sample configuration for the DAC module. The port pin is configured, AVDD is selected for the voltage reference, and the DAC output is enabled.

EXAMPLE 10-1: DAC CONFIGURATION

```
;*  This code block will configure the DAC
;*  for AVDD Voltage Ref, and RB1/AN5/VDAC as
;*  output.

    BANKSEL TRISB          ; Select bank 1
    BSF     TRISB,1        ; Set RB1 input
    BSF     ANSEL,1        ; Set RB1 as analog

    BANKSEL DACON0         ; Select Bank 2
    CLRF    DAC            ; DAC to 00
    MOVLW   B'11000000'    ; Enable DAC output
    MOVWF   DACON0         ; Set REF = VDD

    MOVLW   DAC_VALUE
    MOVWF   DAC            ; Set DAC output
```

10.4 Effects of RESET

A device RESET forces all registers to their RESET state. This forces the following conditions:

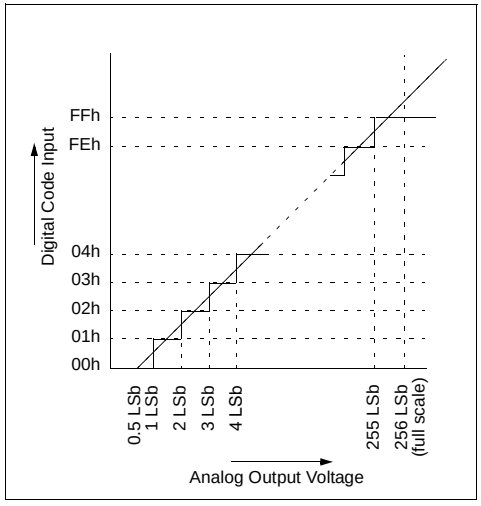
- DAC module is off
- Reference input to AVDD
- Output disabled
- DAC register is cleared

10.5 DAC Module Accuracy/Error

The accuracy/error specified for the DAC includes:

- Integral non-linearity error
- Differential non-linearity error
- Gain error
- Offset error
- Monotonicity

FIGURE 10-2: DAC TRANSFER FUNCTION



Offset error measures the first actual transition of a code versus the first ideal transition of a code. Offset error shifts the entire transfer function. Offset error can be calibrated out of a system or introduced into a system through the interaction of the output drive capability with the load impedance.

Gain error measures the maximum deviation of the last actual transition and the last ideal transition adjusted for offset error. This error appears as a change in slope of the transfer function. The difference in gain error to full scale error is that full scale does not take offset error into account. Gain error can be calibrated out by adjusting the reference voltage.

Linearity error refers to the uniformity of the voltage change with code change. Linearity errors cannot be calibrated out of the system. **Integral non-linearity error** measures the actual voltage output versus the ideal voltage output adjusted by the gain error for each code.

Differential non-linearity error measures the maximum actual voltage step versus the ideal voltage step. This measure is unadjusted.

TABLE 10-1: REGISTERS/BITS ASSOCIATED WITH DAC

Address	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
11Fh	DACON0	DAON	DAOE	—	—	—	—	DARS1	DARS0	00-- --00	00-- --00
11Eh	DAC	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	0000 0000	0000 0000
86h	TRISB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	1111 1111	1111 1111
9Dh	ANSEL	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used for DAC conversion.

NOTES:

11.2 Configuration as OPAMP or Comparator

The following example demonstrates calibration of the OPA module as an Operational Amplifier.

EXAMPLE 11-1: CALIBRATION FOR OPAMP MODE

```
;*  This code block will configure the OPA
;*  module as an Op Amp, 2 MHz GBWP, and
;*  calibrated for a common mode voltage of
;*  1.2V. Routine returns w=0 if
;*  calibration good.

    BANKSEL     OPACON       ; Select Bank 2
    MOVLW       B'10000001'  ; Op Amp mode &
    MOVWF       OPACON       ; 2 MHz GBWP

    BCF         CALCON,CALREF; Set 1.2V
    BSF         CALCON,CAL   ; Start

CAL_LOOP
    BTFSC       CALCON,CAL   ; Test for end
    GOTO        CAL_LOOP     ; If not, wait
    MOVLW       ERROR_FLAG
    BTFSS       CALCON,CALERR; Test for error
    CLRW                     ; If no, return 0
    RETURN
```

The following example demonstrates how to configure and calibrate the OPA module as a Voltage Comparator.

EXAMPLE 11-2: CALIBRATION FOR COMPARATOR MODE

```
;*  This code block will configure the OPA
;*  module as a voltage comparator, slow
;*  speed, and calibrated for a common mode
;*  voltage of 2.5 V (assumes VDD=5V).
;*  Routine returns w=0 if calibration good.

    BANKSEL     OPACON       ; Select Bank 2
    MOVLW       B'10000000'
    MOVWF       OPACON       ; Op Amp mode,
                             ; slow
    BSF         CALCON,CALREF; Common mode=DAC

    MOVLW       H'0x80'
    MOVWF       DAC          ; DAC at VDD/2
    MOVLW       B'10000000'
    MOVWF       DACON0       ; enable DAC,
                             ; VDD ref
    BSF         CALCON,CAL   ; Start
CAL_LOOP
    BTFSC       CALCON,CAL   ; Test for end
    GOTO        CAL_LOOP     ; If not, wait

    MOVLW       ERROR_FLAG
    BTFSS       CALCON,CALERR; Test for error
    CLRW                     ; If no, return 0
    BSF         OPACON,CMPEN ; Comparator mode
    RETURN
```

11.3 Effects of RESET

A device RESET forces all registers to their RESET state. This disables the OPA module and clears any calibration.

11.4 OPA Module Performance

Common AC and DC performance specifications for the OPA module:

- Common Mode Voltage Range
- Leakage Current
- Input Offset Voltage
- Open Loop Gain
- Gain Bandwidth Product

Common mode voltage range is the specified voltage range for the OPA+ and OPA- inputs, for which the OPA module will perform to within its specifications. The OPA module is designed to operate with input voltages between 0 and V_{DD}-1.4V. Behavior for Common mode voltages greater than V_{DD}-1.4V, or below 0V, are not guaranteed.

Leakage current is a measure of the small source or sink currents on the OPA+ and OPA- inputs. To minimize the effect of leakage currents, the effective impedances connected to the OPA+ and OPA- inputs should be kept as small as possible and equal.

Input offset voltage is a measure of the voltage difference between the OPA+ and OPA- inputs in a closed loop circuit with the OPA in its linear region. The offset voltage will appear as a DC offset in the output equal to the input offset voltage, multiplied by the gain of the circuit. The input offset voltage is also affected by the Common mode voltage. The OPA has an automatic calibration module which can minimize the input offset voltage of the module.

Open loop gain is the ratio of the output voltage to the differential input voltage, (OPA+) - (OPA-). The gain is greatest at DC and falls off with frequency.

Gain Bandwidth Product or GBWP is the frequency at which the open loop gain falls off to 0 dB. The lower GBWP is optimized for systems requiring low frequency response and low power consumption.

REGISTER 13-1: PSMC CONTROL REGISTER0 (PSMCCON0: 111h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SMCCL1	SMCCL0	MINDC1	MINDC0	MAXDC1	MAXDC0	DC1	DC0
bit 7							bit 0

bit 7-6 **SMCCL<1:0>:** Clock Frequency Select bits
00 = Output frequency for single output mode is FOSC/128
01 = Output frequency for single output mode is FOSC/64
10 = Output frequency for single output mode is FOSC/32
11 = Output frequency for single output mode is FOSC/16

bit 5-4 **MINDC<1:0>:** Minimum Duty Cycle Select bits for PWM Mode
00 = Min duty cycle of 0
01 = Min duty cycle of 1/8
10 = Min duty cycle of 1/4
11 = Min duty cycle of 3/8

bit 3-2 **MAXDC<1:0>:** Maximum Duty Cycle Select bits for PWM Mode
00 = Max duty cycle of 1/2
01 = Max duty cycle of 5/8
10 = Max duty cycle of 3/4
11 = Max duty cycle of 15/16

bit 1-0 **DC<1:0>:** Duty Cycle Select bits for PSM Mode
00 = Duty cycle of 1/8
01 = Duty cycle of 3/8
10 = Duty cycle of 5/8
11 = Duty cycle of 15/16

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

13.3.2 EXAMPLE BUCK LC SWITCHING POWER SUPPLY

In this example, the PSMC controls the buck configuration switching power supply in Figure 13-6.

The PSMC is configured as a typical PWM, current mode, switching power supply controller. The inner current feedback loops consist of:

- PSMC
- 2 MOSFET drivers
- Power MOSFETs Q1 and Q2
- Inductors L1 and L2
- Current transformer
- Comparator C1/C2

The outer voltage feedback loop consists of:

- Diodes D1, D2, D3, and D4
- CMAIN
- OPAMP feedback filter
- DAC reference

The circuit uses two feedback loops, an inner current control loop, and an outer voltage loop. The inner loop is further divided into two channels, Q1/L1, and Q2/L2. The PSMC operates a PWM output, alternately driving Q1 for a cycle, then driving Q2 the next. During the active phase of either output pulse, the inner loop builds up a current flow in the output's inductor, proportional to the error voltage received from the OPAMP. The current flow in the inductor begins the charging of CMAIN. When the voltage (proportional to the current flow in the inductor) exceeds the error voltage:

- The comparator resets the PSMC output
- The MOSFET is turned off
- The flyback diode forward biases
- The inductor discharges into CMAIN for the remainder of the period.

The outer voltage loop monitors the output voltage across CMAIN via R1/R2. The reference voltage from the DAC is subtracted from the feedback voltage to generate the raw error voltage. The raw error voltage is then filtered by the OPAMP and routed to Comparator C1 in the inner current loop.

In using two alternating outputs, the outputs are limited to less than 50% duty cycle. As a result, the circuit avoids the problems associated with instability at duty cycles of >50%.

For more information concerning the design of switching power supplies, refer to:

Switching Power Supply Design, by Abraham I. Pressman, published by McGraw Hill (ISBN 0-07-052236-7).

Note:	Following RESET, both the PSMC1A and PSMC1B outputs are held tri-state until the PSMC is configured. Driver circuitry for all power MOSFET transistors must have a resistor bias to turn off the transistor in the event of tri-state conditions on either output to prevent undo stress on the MOSFET's and their associated circuitry.

15.0 INSTRUCTION SET SUMMARY

Each PIC16CXXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXXX instruction set summary in Table 15-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 15-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

TABLE 15-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
TO	Time-out bit
PD	Power-down bit

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 15-2 lists the instructions recognized by the MPASM™ assembler.

Figure 15-1 shows the general formats that the instructions can have.

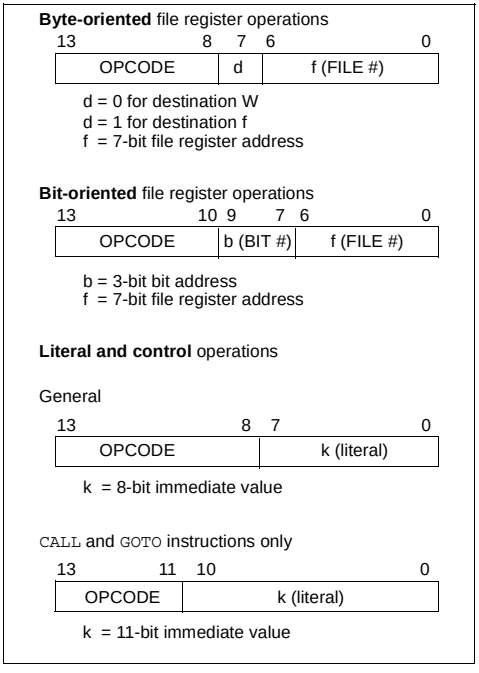
Note: To maintain upward compatibility with future PIC16CXXX products, do not use the OPTION and TRIS instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 15-1: GENERAL FORMAT FOR INSTRUCTIONS



A description of each instruction is available in the PIC Mid-Range Reference Manual, (DS33023).

TABLE 17-9: AC CHARACTERISTICS: OPERATIONAL AMPLIFIER (OPA)

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated): V_{DD} = 2.7V to 5.5V, V_{SS} = GND, T_A = 25°C, VCM = V_{DD}/2, RL = 100kΩ to V_{DD}/2, and V_{OUT} = V_{DD}/2 Operating Temperature -40°C to +85°C for Industrial				
Param No.	**Parameters**	**Symbol**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	Gain Bandwidth Product	GBWP	—	75	—	kHz	V_{DD} = 5V, GBWP = 0
		GBWP	—	2	—	MHz	V_{DD} = 5V, GBWP = 1
	Input Offset Auto	T_Z	—	300	TBD	µs	V_{DD} = 5V, GBWP = 1
	Calibration Time	T_Z	—	TBD	TBD	µs	V_{DD} = 5V GBWP = 0
		T_{ON}	—	10	TBD	µs	V_{DD} = 5V, GWBP = 1
	Turn On Time	T_{ON}	—	TBD	TBD	µs	V_{DD} = 5V, GBWP = 0
	Phase Margin	ΘM	—	TBD	—	degrees	V_{DD} = 5V, GBWP = 0
		ΘM	—	TBD	—	degrees	V_{DD} = 5V, GBWP = 1
	Slew Rate	SR	—	TBD	—	V/µs	V_{DD} = 5V, GBWP = 0
		SR	—	TBD	—	V/µs	V_{DD} = 5V, GBWP = 1

Note: Data in 'Typ' column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

T

V

W

