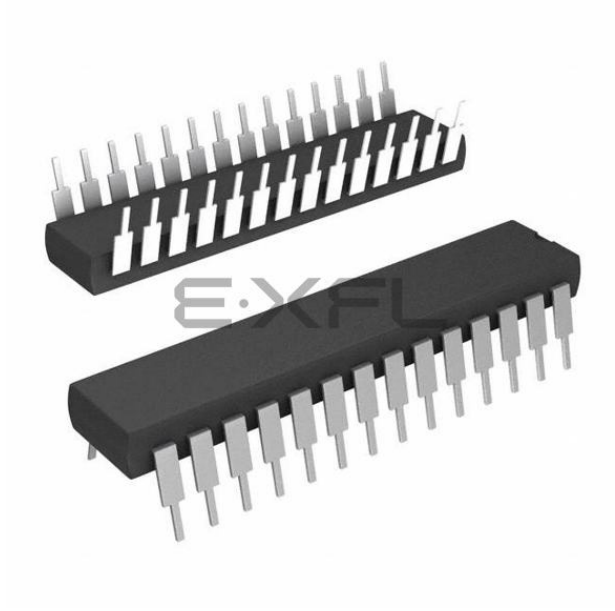


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C
Peripherals	POR, Temp Sensor, WDT
Number of I/O	20
Program Memory Size	7KB (4K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6V
Data Converters	Slope A/D
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic14000-20-sp

NOTES:

TABLE 4-3: SPECIAL FUNCTION REGISTERS FOR THE PIC14000 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bank1									
80h*	INDF (Indirect Address)	Addressing this location uses contents of FSR to address data memory (not a physical register).							
81h	OPTION	$\overline{RCPU}$	r	TOCS	TOSE	PSA	PS2	PS1	PS0
82h*	PCL	Program Counter's (PC's) least significant byte							
83h*	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
84h*	FSR	Indirect data memory address pointer							
85h	TRISA	PORTA Data Direction Register							
86h	Reserved	Reserved for emulation							
87h	TRISC	PORTC Data Direction Register							
88h	TRISD	PORTD Data Direction Register							
89h	Reserved								
8Ah*	PCLATH	Buffered register for the upper 5 bits of the Program Counter (PC)							
8Bh*	INTCON	GIE	PEIE	T0IE	r	r	T0IF	r	r
8Ch	PIE1	CMIE	—	—	PBIE	I^2CIE	RCIE	ADCIE	OVFIE
8Dh	Reserved								
8Eh	PCON	r	—	—	—	—	—	$\overline{POR}$	$\overline{LVD}$
8Fh	SLPCON	HIBEN	—	REFOFF	LSOFF	OSCOFF	CMOFF	TEMPOFF	ADOFF
90h	Reserved								
91h	Reserved								
92h	Reserved								
93h	I^2CADD	I^2C Synchronous Serial Port Address Register							
94h	I^2CSTAT	—	—	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
95h	Reserved								
96h	Reserved								
97h	Reserved								
98h	Reserved								
99h	Reserved								
9Ah	Reserved								
9Bh	PREFA	PRA7	PRA6	PRA5	PRA4	PRA3	PRA2	PRA1	PRA0
9Ch	PREFB	PRB7	PRB6	PRB5	PRB4	PRB3	PRB2	PRB1	PRB0
9Dh	CMCON	—	CMBOUT	CMBOE	CPOLB	—	CMAOUT	CMAOE	CPOLA
9Eh	MISC	SMHOG	SPGNDB	SPGNDA	I^2CSEL	SMBUS	INCLKEN	OSC2	OSC1
9Fh	ADCON1	ADDAC3	ADDAC2	ADDAC1	ADDAC0	PCFG3	PCFG2	PCFG1	PCFG0

Legend
— = unimplemented bits, read as '0' but cannot be overwritten
r = reserved bits, default is POR value and **should not be overwritten with any value**
Reserved indicates reserved register and **should not be overwritten with any value**
* indicates registers that can be addressed from either bank

4.4 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

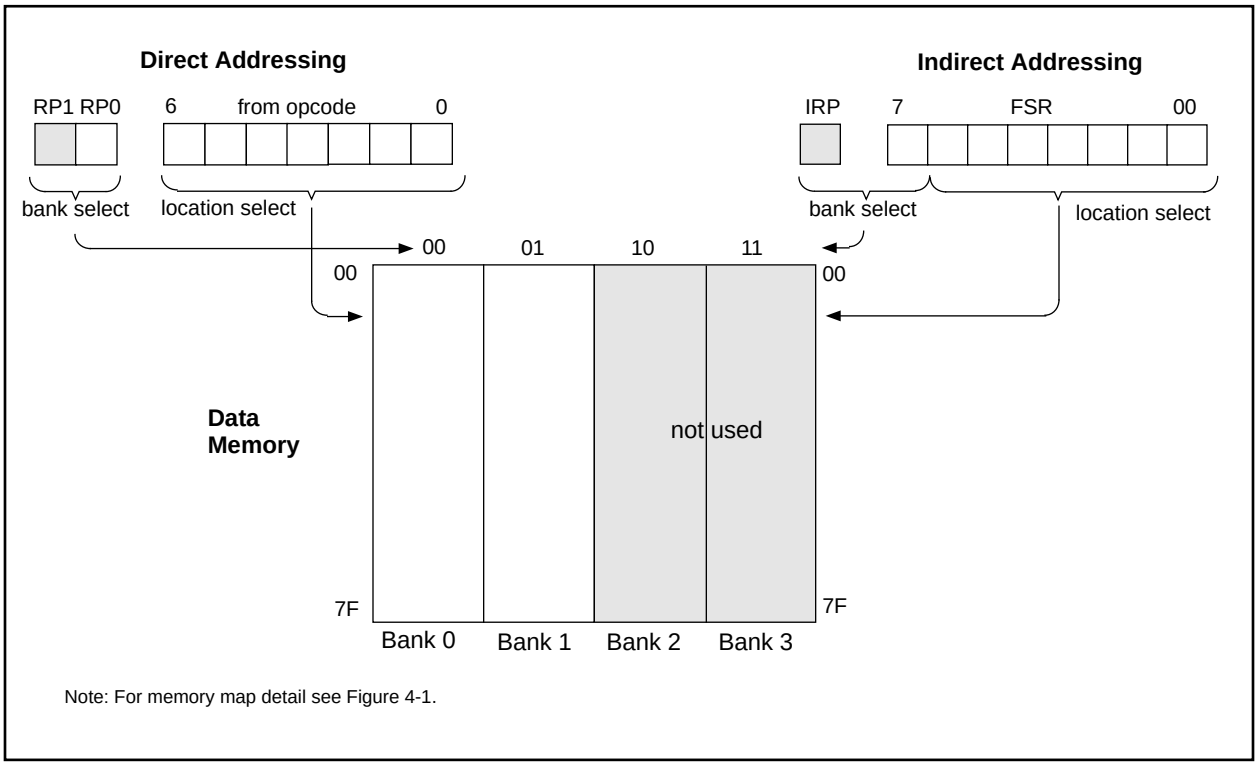
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the file select register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 4-10. However, IRP is not used in the PIC14000.

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 4-2.

EXAMPLE 4-2: INDIRECT ADDRESSING

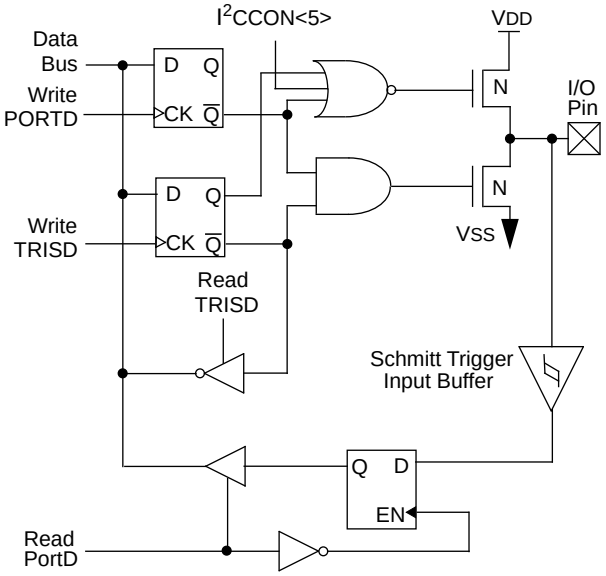
```
        movlw   0x20    ;initialize pointer
        movf    FSR     ;to RAM
NEXT    clrf    INDF    ;clear INDF register
        incf    FSR     ;inc pointer
        btfss   FSR,4   ;all done?
        goto    NEXT    ;no clear next
                        ;yes continue
CONTINUE:
```

FIGURE 4-10: INDIRECT/INDIRECT ADDRESSING



Note: For memory map detail see Figure 4-1.

FIGURE 5-10: BLOCK DIAGRAM OF PORTD<1:0> PINS



Note: I/O pins have protection diodes to VDD and VSS. These pins do not have a P-channel pull-up.

FIGURE 5-11: PORTD DATA REGISTER

08h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTD	RD7/AN7	RD6/AN6	RD5/AN5	RD4/AN4	RD3/REFB	RD2/CMPB	RD1/SDAB	RD0/SCLB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR value xxh	X	X	X	X	X	X	X	X

Bit	Name	Function
B7	RD7/AN7	GPIO or analog input. Returns value on pin RD7/AN7 when used as a digital input. When configured as an analog input, reads as '0'.
B6	RD6/AN6	GPIO or analog input. Returns value on pin RD6/AN6 when used as a digital input. When configured as an analog input, reads as '0'.
B5	RD5/AN5	GPIO or analog input. This pin can connect to a level shift network. If enabled, a +0.5V offset is added to the input voltage. When configured as an analog input, reads as '0'.
B4	RD4/AN4	GPIO or analog input. Returns value on pin RD4/AN4 when used as a digital input. When configured as an analog input, reads as '0'.
B3	RD3/REFB	This pin can serve as a GPIO, or programmable reference B output.
B2	RD2/CMPB	This pin can serve as a GPIO, or comparator B output.
B1	RD1/SDAB	Alternate synchronous serial data I/O for I^2C interface enabled by setting the I^2CSEL bit in the MISC register. This pin can also serve as a general purpose I/O. This pin has an N-channel pull-up to VDD which is disabled in I^2C mode.
B0	RD0/SCLB	Alternate synchronous serial clock for I^2C interface, enabled by setting the I^2CSEL bit in the MISC register. This pin can also serve as a general purpose I/O. This pin has an N-Channel pull-up to VDD which is disabled in I^2C mode.

Legend: U = unimplemented, read as '0', x = unknown.

6.2 Using Timer0 with External Clock

When the external clock input (pin RC3/T0CKI) is used for Timer0, it must meet certain requirements. The external clock requirement is due to internal phase clock (TOSC) synchronization. Also, there is a delay in the actual incrementing of TMR0 after synchronization.

6.2.1 EXTERNAL CLOCK SYNCHRONIZATION

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-5). Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns).

When a prescaler is used, the external clock input is divided by the asynchronous ripple counter-type prescaler so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4Tosc (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns.

6.2.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 6-5 shows the delay from the external clock edge to the timer incrementing.

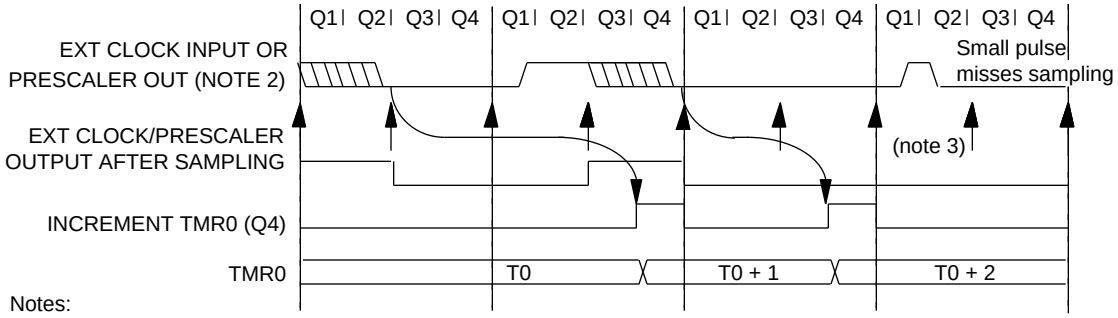
6.3 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a post-scaler for the Watchdog Timer (Figure 6-1). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available which is mutually exclusive between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

Bit PSA and PS2:PS0 (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the Timer0 module (e.g., `CLRF 1`, `MOVWF 1`, `BSF 1,x`) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

FIGURE 6-5: TIMER0 TIMING WITH EXTERNAL CLOCK



Notes:

1. Delay from clock input change to TMR0 increment is 3 TOSC to 7 TOSC. (Duration of Q = TOSC). Therefore, the error in measuring the interval between two edges on TMR0 input = ± 4 tosc max.
2. External clock if no prescaler selected, Prescaler output otherwise.
3. The arrows indicate the points in time where sampling occurs.

7.5.1.1 ADDRESSING

Once the I^2C module has been enabled, the I^2C waits for a START to occur. Following the START, the 8-bits are shifted into the I^2CSR. All incoming bits are sampled with the rising edge of the clock (SCL) line. The $I^2CSR<7:1>$ is compared to the I^2CADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match, and the BF and I^2COV bits are clear, the following things happen:

- I^2CSR loaded into I^2CBUF
- Buffer Full (BF) bit is set
- $\overline{ACK}$ pulse is generated
- I^2C Interrupt Flag (I^2CIF) is set (interrupt is generated if enabled (I^2CIE set) on falling edge of ninth SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave (Figure 7-5). The five most significant bits (MSbs) of the first address byte specify if this is a 10-bit address. The R/$\overline{W}$ bit (bit 0) must specify a write, so the slave device will received the second address byte. For a 10-bit address the first byte would equal '1 1 1 1 0 A9 A8 0', where A9 and A8 are the two MSbs of the address. The sequence of events for 10-bit address are as follows, with steps 7-9 for slave-transmitter:

1. Receive first (high) byte of address (I^2CIF, BF and UA are set).
2. Update I^2CADD with second (low) byte of address (clears UA and releases SCL line).
3. Read I^2CBUF (clears BF) and clear I^2CIF.
4. Receive second (low) byte of address (I^2CIF, BF and UA are set).
5. Update I^2CADD with first (high) byte of address (clears UA, if match releases SCL line).
6. Read I^2CBUF (clears BF) and clear I^2CIF
7. Receive Repeated START.
8. Receive first (high) byte of address (I^2CIF and BF are set).
9. Read I^2CBUF (clears BF) and clear I^2CIF.

7.5.1.2 RECEPTION

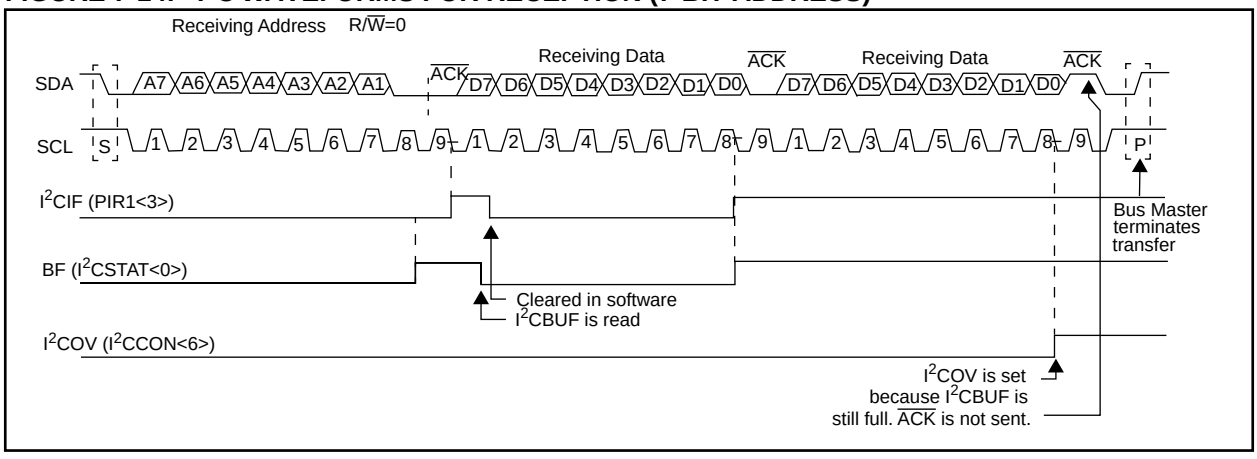
When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the I^2CSTAT register is cleared. The received address is loaded into the I^2CBUF.

When the address byte overflow condition exists then no acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either the BF bit ($I^2CSTAT<0>$) is set or the I^2COV bit ($I^2CCON<6>$) is set (Figure 7-14).

An I^2CIF interrupt is generated for each data transfer byte. The I^2CIF bit must be cleared in software, and the I^2CSTAT register is used to determine the status of the byte. In master mode with slave enabled, three interrupt sources are possible. Reading BF, P and S will indicate the source of the interrupt.

Caution: BF is set after receipt of eight bits and automatically cleared after the I^2CBUF is read. However, the flag is not actually cleared until receipt of the acknowledge pulse. Otherwise extra reads appear to be valid.

FIGURE 7-14: I^2C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)



7.5.2 MASTER MODE

Master mode operation is supported by interrupt generation on the detection of the START and STOP. The STOP(P) and START(S) bits are cleared from a reset or when the I^2C module is disabled. Control of the I^2C bus may be taken when the P bit is set, or the bus is idle and both the S and P bits are cleared.

In master mode, the SCL and SDA lines are manipulated by changing the corresponding TRISC<7:6> or TRISD<1:0> bits to an output (cleared). The output level is always low, regardless of the value(s) in PORTC<7:6> or PORTD<1:0>. So when transmitting data, a “1” data bit must have the TRISC<7> or TRISD<1> bit set (input) and a “0” data bit must have the TRISC<7> or TRISD<1> bit cleared (output). The same scenario is true for the SCL line with the TRISC<6> or TRISD<0> bit.

The following events will cause the I^2C interrupt Flag (I^2CIF) to be set (I^2C interrupt if enabled):

- START
- STOP
- Data transfer byte transmitted/received

Master mode of operation can be done with either the slave mode idle ($I^2CM3...I^2CM0$ = 1011b) or with the slave active. When both master and slave modes are enabled, the software needs to differentiate the source(s) of the interrupt.

7.5.3 MULTI-MASTER MODE

In multi-master mode, the interrupt generation on the detection of the START and STOP allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a reset or when the I^2C module is disabled. Control of the I^2C bus may be taken when the P bit is set, or the bus is idle and both the S and P bits are cleared. When the bus is busy, enabling the I^2C interrupt will generate the interrupt when the STOP occurs.

In multi-master operation, the SDA line must be monitored to see if the signal level is the expected output level. This check only needs to be done when a high level is output. If a high level is expected and low level is present, the device needs to release the SDA and SCL lines (set TRISC<7:6>). There are two stages where this arbitration can be lost, these are:

- Address Transfer
- Data Transfer

When the slave logic is enabled, the slave continues to receive. If arbitration was lost during the address transfer stage, the device may being addressed. If addressed an $\overline{ACK}$ pulse will be generated. If arbitration was lost during the data transfer stage, the device will need to re-transfer the data at a later time.

TABLE 7-3: REGISTERS ASSOCIATED WITH I^2C OPERATION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0B/8Bh	INTCON	GIE	PEIE	T0IE	r	r	T0IF	r	r
0Ch	PIR1	CMIF	—	—	PBIF	I^2CIF	RCIF	ADCIF	OVFIF
8Ch	PIE1	CMIE	—	—	PBIE	I^2CIE	RCIE	ADCIE	OVFIE
13h	I^2CBUF	I^2C Serial Port Receive Buffer/Transmit Register							
93h	I^2CADD	I^2C mode Synchronous Serial Port (I^2C mode) Address Register							
14h	I^2CCON	WCOL	I^2CON	I^2CEN	CKP	I^2CM3	I^2CM2	I^2CM1	I^2CM0
94h	I^2CSTAT	—	—	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
9Eh	MISC	SMHOG	SPGNDB	SPGNDA	I^2CSEL	SMBUS	INCLKEN	OSC2	OSC1
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
88h	TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0

Legend: — = Unimplemented location, read as ‘0’
r = reserved locations, default is POR value and **should not be overwritten with any value**

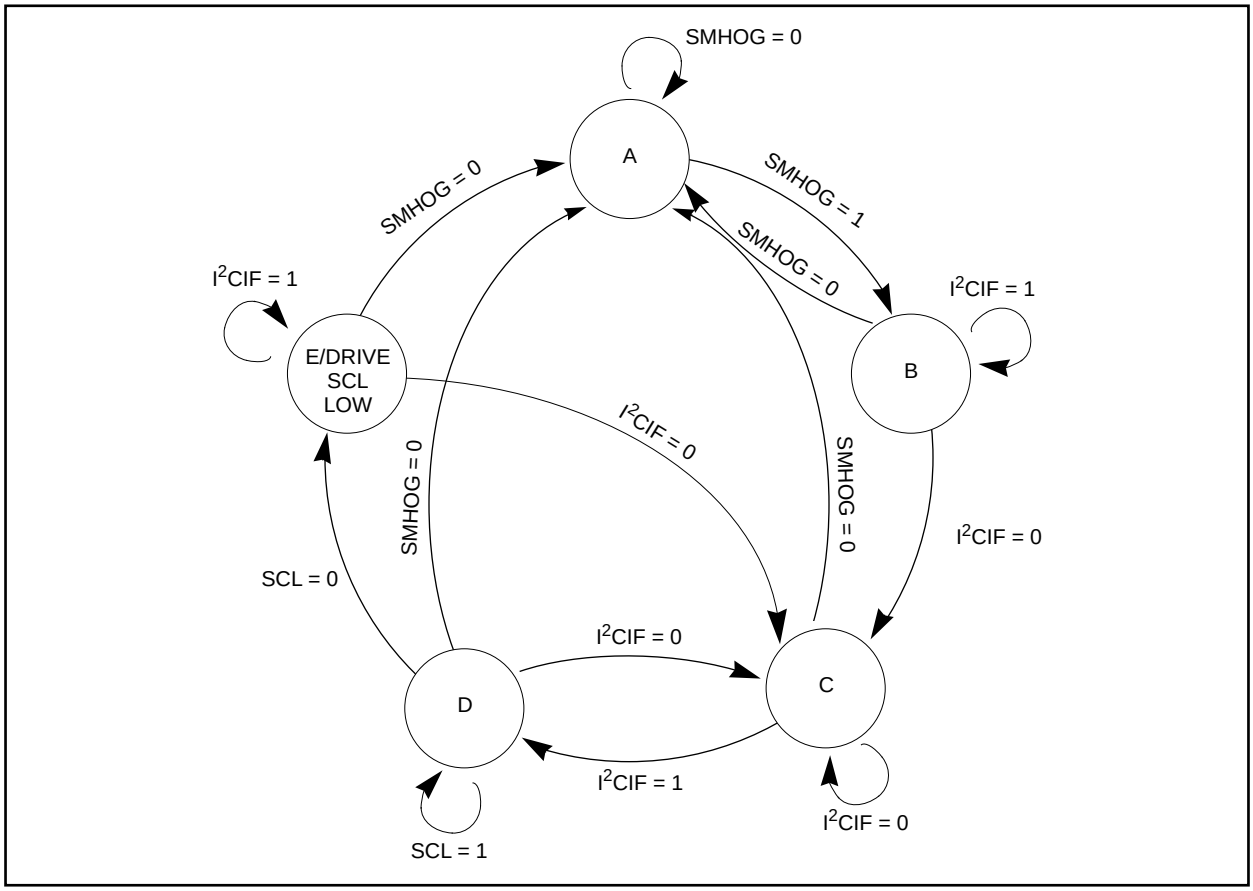
Note: Shaded boxes are not used by the I^2C module.

7.5.4 SMBus™ AND ACCESS.bus™ CONSIDERATIONS

PIC14000 is compliant with the SMBus specification published by Intel. Some key points to note regarding the bus specifications and how it pertains to the PIC14000 hardware are listed below:

- SMBus has fixed input voltage thresholds. PIC14000 I/O buffers have programmable levels that can be selected to be compatible with both SMBus threshold levels via the SMBus and SPGND bits in the MISC register.
- A mechanism to stretch the I^2C clock time has been implemented to support SMBus slave transactions. The SMHOG bit (MISC<7>) allows hardware to automatically force and hold the I^2C clock line low when a data byte has been received. This prevents the SMBus master from overflowing the receive buffer in instances where the microcontroller may be to busy servicing higher priority tasks to respond to a I^2C module interrupt. Or, if the microcontroller is in SLEEP mode and needs time to wake-up and respond to the I^2C interrupt.

FIGURE 7-18: SMHOG STATE MACHINE



10.6 Interrupts

The PIC14000 has several sources of interrupt:

- External interrupt from OSC1/PBTN pin
- I^2C port interrupt
- PORTC interrupt on change (pins RC<7:4> only)
- Timer0 overflow
- A/D timer overflow
- A/D converter capture event
- Programmable reference comparator interrupt

This section addresses the external and Timer0 interrupts only. Refer to the appropriate sections for description of the serial port, programmable reference and A/D interrupts.

INTCON records individual interrupt requests in flag bits. It also has individual and global enable bits. The peripheral interrupt flags reside in the PIR1 register. Peripheral interrupt enable interrupts are contained in the PIE1 register.

Global interrupt masking is controlled by GIE (INTCON<7>). Individual interrupts can be disabled through their corresponding mask bit in the INTCON register. GIE is cleared on reset to mask interrupts.

When an interrupt is serviced, the GIE is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h, the interrupt vector. For external interrupt events, such as the I^2C interrupt, the interrupt latency will be 3 or 4 instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for 1 or 2 cycle instructions. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid infinite interrupt requests. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit to allow polling.

The return from interrupt instruction, RETFIE, exits the interrupt routine as well as sets the GIE bit to re-enable interrupts.

Note 1: The individual interrupt flags will be set by the specified condition even though the corresponding interrupt enable bit is cleared (interrupt disabled) or the GIE bit is cleared (all interrupts disabled).

Note 2: If an interrupt occurs while the Global Interrupt Enable (GIE) bit is being cleared, the GIE bit may unintentionally be re-enabled by the user's Interrupt Service Routine (the RETFIE instruction). The events that would cause this to occur are:

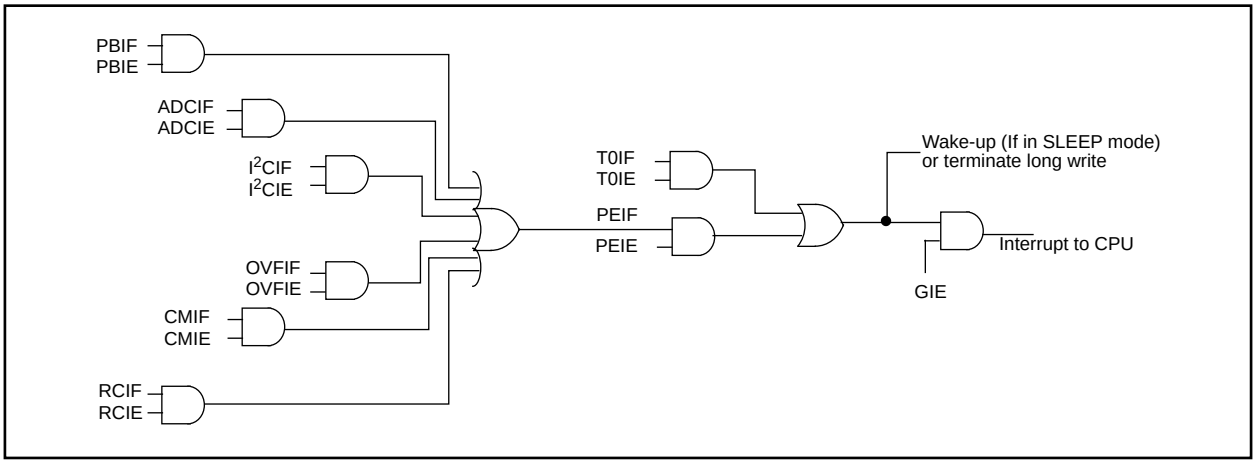
1. An instruction clears the GIE bit while an interrupt is acknowledged.
2. The program branches to the interrupt vector and executes the Interrupt Service Routine.
3. The interrupt service routine completes with the execution of the RETFIE instruction. This causes the GIE bit to be set (enables interrupts), and the program returns to the instruction after the one which was meant to disable interrupts.

The method to ensure that interrupts are globally disabled is:

1. Ensure that the GIE bit was cleared by the instruction, as shown in the following code:

```
LOOP: BCF   INTCON,GIE  ; Disable Global Interrupts
      BTFSC INTCON,GIE  ; Global Interrupts Disabled?
      GOTO  LOOP        ; No, try again
      :                 ; Yes, continue with program
                        ;     flow
```

FIGURE 10-9: INTERRUPT LOGIC SCHEMATIC



10.7.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms (with no prescaler). The time-out periods vary with temperature, VDD and process variations (see DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION registers. Thus, time-out periods up to 2.3 seconds can be realized. The CLRWDT and SLEEP instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET.

The $\overline{TO}$ bit in the status register will be cleared upon a watchdog timer time-out. The WDT time-out period (no prescaler) is measured and stored in calibration space at location 0FD2h.

10.7.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken into account that under worst-case conditions (minimum VDD, maximum temperature, maximum WDT prescaler) it may take several seconds before a WDT time-out occurs. Refer to Section 6.3 for prescaler switching considerations.

10.8 Power Management Options

The PIC14000 has several power management options to prolong battery lifetime. The SLEEP instruction halts the CPU and can turn off the on-chip oscillators. The CPU can be in SLEEP mode, yet the A/D converter can continue to run. Several bits are included in the SLPCON register (8Fh) to control power to analog modules.

TABLE 10-6: SUMMARY OF POWER MANAGEMENT OPTIONS

Function	Summary
CPU Clock	OFF during SLEEP/HIBERNATE mode, ON otherwise
Main Oscillator	ON if NOT in SLEEP mode. In SLEEP mode, controlled by OSCOFF bit, SLPCON<3>.
Watchdog Timer	Controlled by WDTE, 2007h<2> and HIBEN, SLPCON<7>
Temperature Sensor	Controlled by TEMPOFF, SLPCON<1>
Low-voltage Detector	Controlled by REFOFF, SLPCON<5>
Comparator and Programmable References	Controlled by CMOFF, SLPCON<2>
A/D Comparator	Controlled by ADOFF, SLPCON<0>
Programmable Current Source	Controlled by ADOFF, SLPCON<0> and ADCON1<7:4>
Slope Reference Voltage Divider	Controlled by ADOFF, SLPCON<0>
Level Shift Networks	Controlled by LSOFF, SLPCON<4>
Bandgap Reference	Controlled by REFOFF, SLPCON<5>
Voltage Regulator Control	Always ON. Does not consume power if unconnected.
Power On Reset	Always ON, except in SLEEP/HIBERNATE mode

Note: Refer to analog specs for individual peripheral operating currents.

11.0 INSTRUCTION SET SUMMARY

The PIC14000's instruction set is the same as PIC16CXX. Each 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 instruction set summary in Table 11-2 lists byte-oriented, bit-oriented, and literal and control operations. Table 11-1 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 11-1: 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
label	Label name
TOS	Top of Stack
PC	Program Counter
PCLATH	Program Counter High Latch
GIE	Global Interrupt Enable bit
WDT	Watchdog Timer/Counter
$\overline{TO}$	Time-out bit
$\overline{PD}$	Power-down bit
dest	Destination either the W register or the specified register file location
[ ]	Options
( )	Contents
$\rightarrow$	Assigned to
< >	Register bit field
$\in$	In the set of
italics	User defined term (font is courier)

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 11-2 lists the instructions recognized by the MPASM assembler.

Figure 11-1 shows the three general formats that the instructions can have.

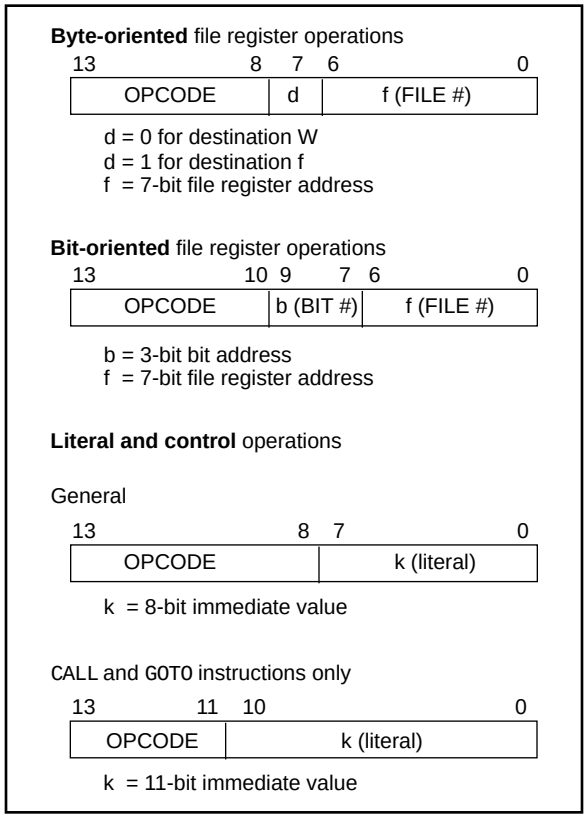
Note: To maintain upward compatibility with future PIC16CXX products, do not use the OPTION and TRIS instructions.

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

0xhh

where h signifies a hexadecimal digit.

FIGURE 11-1: GENERAL FORMAT FOR INSTRUCTIONS



GOTO **Unconditional Branch**

Syntax: [*label*] GOTO k

Operands: $0 \le k \le 2047$

Operation: $k \rightarrow PC<10:0>$
$PCLATH<4:3> \rightarrow PC<12:11>$

Status Affected: None

Encoding:

10	1kkk	kkkk	kkkk

Description: GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction.

Words: 1

Cycles: 2

Example
```
GOTO THERE
```
After Instruction
PC = Address THERE

INCF **Increment f**

Syntax: [*label*] INCF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) + 1 → (dest)

Status Affected: Z

Encoding:

00	1010	dfff	ffff

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

Words: 1

Cycles: 1

Example
```
INCF CNT, 1
```
Before Instruction
CNT = 0xFF
Z = 0
After Instruction
CNT = 0x00
Z = 1

INCFSZ **Increment f, Skip if 0**

Syntax: [*label*] INCFSZ f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) + 1 → (dest), skip if result = 0

Status Affected: None

Encoding:

00	1111	dfff	ffff

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.
If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two cycle instruction.

Words: 1

Cycles: 1(2)

Example
```
HERE     INCFSZ   CNT, 1
         GOTO     LOOP
CONTINUE •
         •
         •
```
Before Instruction
PC = address HERE
After Instruction
CNT = CNT + 1
if CNT= 0,
PC = address CONTINUE
if CNT≠ 0,
PC = address HERE +1

IORLW **Inclusive OR Literal with W**

Syntax: [*label*] IORLW k

Operands: $0 \le k \le 255$

Operation: (W) .OR. k → (W)

Status Affected: Z

Encoding:

11	1000	kkkk	kkkk

Description: The contents of the W register is OR'ed with the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Example
```
IORLW  0x35
```
Before Instruction
W = 0x9A
After Instruction
W = 0xBF
Z = 1

13.3 Timing Parameter Symbology

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS
3. TCC:ST (I^2C specifications only)
4. Ts (I^2C specifications only)

T			
F	Frequency	T	Time

Lowercase subscripts (pp) and their meanings:

pp			
ck	CLKOUT	osc	OSC1
di	SDI	t0	T0CKI
io	I/O port		
mc	$\overline{MCLR}$		

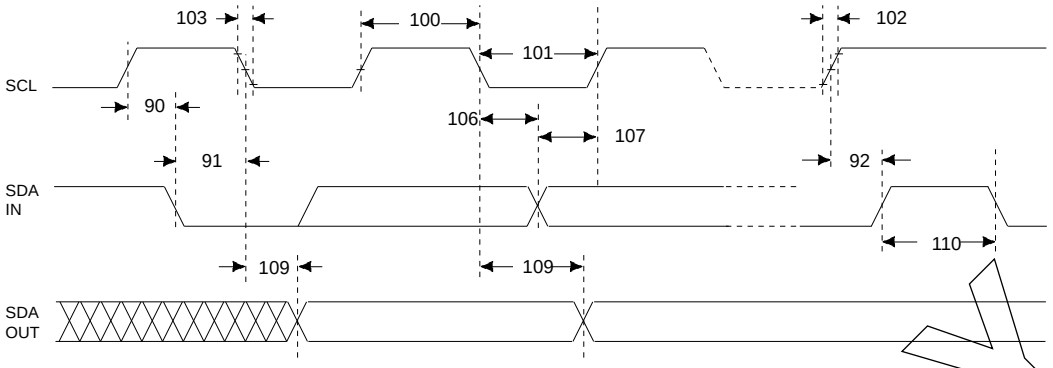
Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
I^2C only			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I^2C specifications only)

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 13-7: I^2C BUS DATA TIMING



Note: Refer to Figure 13-2 for load conditions

TABLE 13-8: I^2C BUS DATA REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Max	Units	Conditions
100	THIGH	Clock high time	100 kHz mode	4.0	—	µs	PIC14000 must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	PIC14000 must operate at a minimum of 10 MHz
			I^2C Module	1.5 TCY	—		
101	TLOW	Clock low time	100 kHz mode	4.7	—	µs	PIC14000 must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	PIC14000 must operate at a minimum of 10 MHz
			I^2C Module	1.5 TCY	—		
102	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20+0.1 C_b	300	ns	C_b is specified to be from 10-400 pF
103	TF	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20+0.1 C_b	300	ns	C_b is specified to be from 10-400 pF
90	TSU:STA	START condition setup time	100 kHz mode	4.7	—	µs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	µs	
91	THD:STA	START condition hold time	100 kHz mode	4.0	—	µs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	µs	
106	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	µs	
107	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	µs	
			400 kHz mode	0.6	—	µs	
109	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110	TBUF	Bus free time	100 kHz mode	4.7	—	µs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
	C_b	Bus capacitive loading		—	400	pF	

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of STARTs or STOPs.

2: A fast-mode I^2C-bus device can be used in a standard-mode I^2C-bus system, but the requirement tSU:DAT≥250ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line TR max.+tSU:DAT=1000+250=1250 ns (according to the standard-mode I^2C bus specification) before the SCL line is released.

Standard Operating Conditions (unless otherwise stated)

Operating Temperature: -40°C ≤ TA ≤ +85°C for industrial
0°C ≤ TA ≤ +70°C for commercial

VDD range: 2.7V (min) to 6.0V (max) unless otherwise stated.

Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions	Notes
Temperature Sensor (continued)							
Output Linearity	lin(temp)	—	TBD	—			1
Operating Current (sensor on)	idd(temp)	—	150	250	µA	TEMPOFF = 0	2
Operating Current (sensor off)	idd(temp)	—	0	—	µA	TEMPOFF = 1	2
Slope Reference Voltage Divider							
Output Voltage (SREFHI)	voh(sref)	1.14	1.19	1.24	V		
Output Voltage (SREFLO)	vol(sref)	0.10	0.13	0.16	V		
Slope Reference Calibration Factor	KREF	0.09	0.126	0.16		TA = 25°C, VDD = 5V	
KREF Supply Sensitivity	ss(KREF)	—	0.02	—	%/V	From VDDmin to VDDmax	1
KREF Temperature Coefficient	tc(KREF)	—	20	—	ppm/°C	From Tmin to Tmax	1
Operating Current (A/D on)	idd(sref)	—	55	85	µA	ADOFF = 0	2
Operating Current (A/D off)	idd(sref)	—	0	—	µA	ADOFF = 1	2
A/D Comparator							
Input Offset Voltage	ioff(adc)	-10	2	10	mV	Measured over common-mode range	
Input Common Mode Voltage Range	cmr(adc)	0	—	VDD-1.4	V		
Differential Voltage Gain	gain(adc)	—	100	—	dB		1
Common Mode Rejection Ratio	cmrr(adc)	—	80	—	dB	VDD = 5V, TA = 25°C, over common-mode range	1
Power Supply Rejection Ratio	psrr(adc)	—	70	—	dB	TA = 25°C, VDDmin to VDDmax	1
Operating Current (A/D on)	idd(adc)	—	40	65	µA	ADOFF = 0	2
Operating Current (A/D off)		—	0	—	µA	ADOFF = 1	2
Programmable Reference(s)							
Upper Range						TA = 25°C	
Output Voltage	vo(pref)	0.627	0.792	0.957	V	PREFx<7:0> = 7Fh (127 decimal), max	
		0.418	0.528	0.638	V	PREFx<7:0> = 50h (80 decimal), min	
Coarse Resolution	resc(pref)	38.0	48.0	58.0	mV	PREFx<2:0> = constant	
Fine Resolution	resf(pref)	4.0	5.0	6.0	mV	PREFx<7:3> = constant	
Middle Range						TA = 25°C	
Output Voltage	vo(pref)	0.414	0.523	0.632	V	PREFx<7:0> = 4F (79 decimal), max	
		0.380	0.480	0.580	V	PREFx<7:0> = 00h (default), mid-point	
		0.342	0.432	0.522	V	PREFx<7:0> = C8h (200 decimal), min	
Coarse Resolution	resc(pref)	3.8	4.8	5.8	mV	PREFx<2:0> = constant	
Fine Resolution	resf(pref)	0.38	0.46	0.54	mV	PREFx<7:3> = constant	1

Standard Operating Conditions (unless otherwise stated)

Operating Temperature: -40°C ≤ TA ≤ +85°C for industrial
0°C ≤ TA ≤ +70°C for commercial

VDD range: 2.7V (min) to 6.0V (max) unless otherwise stated.

Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions	Notes
Calibration Accuracy						All parameters calibrated at VDD = 5V, TA = 25°C unless noted.	3, 5
				Accuracy			
Parameter	**Sym.**	**Resolution**	**Units**	**Typ**	**Max**	**Conditions**	**Notes**
Slope Reference Ratio	KREF	0.015%	—	.02%	—		
Bandgap Reference Voltage	KBG	10	µV	.01%	—		
Temperature Sensor Output Voltage	VTHERM	20	µV	.02%	—		
Temperature Sensor Slope Coefficient	KTC	0.33	µV/°C	6.7%	—	Calibrated at 25°C and Tmax	
Internal Oscillator Frequency	FOSC	10.0	kHz	0.14%	—		
Watchdog Timer Time-out Period	TWDT	1	ms	0.5 ms	—		

Notes for the analog specifications:

Note 1: This parameter is characterized but not tested.

Note 2: IDD values of individual analog module cannot be tested independently but are characterized.

Note 3: Calibration temp accuracy is ± 1°C typical, ± 2°C max.

Note 4: Guaranteed by design.

Note 5: Refer to AN621 for further information on calibration parameters and accuracy.

Calculations:

Temperature coefficients are calculated as:

tc = (value @TMAX - value @TMIN) / ((TMAX-TMIN) * Average(value @TMAX,value @TMIN))

Temperature coefficient for the internal temperature sensor is calculated as:

tc sensor = (sensor voltage @ TMAX - sensor voltage @ 25°C) / (TMAX - 25°C)

Temperature coefficients for the bandgap reference and programmable current source are calculated as

the larger TC from 25°C to either TMIN or TMAX

Supply sensitivities are calculated as:

ss = (value@VDDMAX - value@VDDMIN)/((VDDMAX - VDDMIN)*
Average(value@VDDMAX, value@VDDMIN))

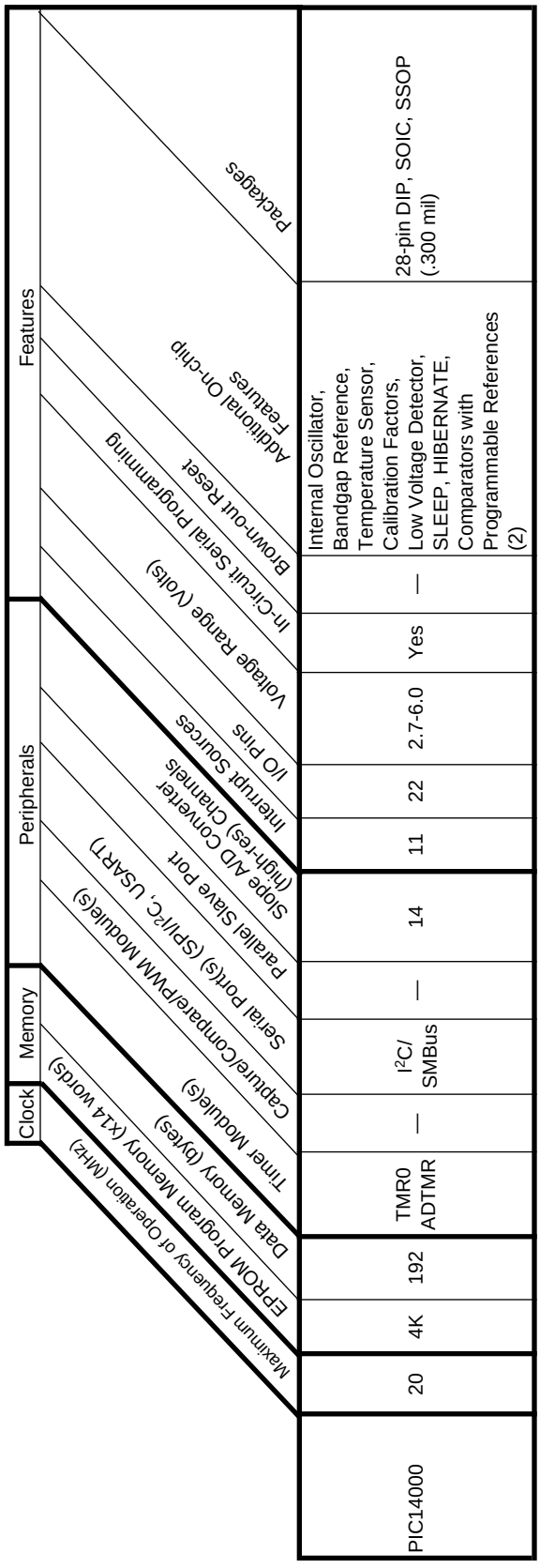
Programmable current source output sensitivity is calculated as:

vs = (value@(VDD - 1.4V) - value @ 0V)/(VDD - 1.4V) *
Average(value@(VDD - 1.4V), value @ 0V)

NOTES:

APPENDIX A: PIC16/17 MICROCONTROLLERS

A.1 PIC14000 Devices



A.8 PIC17CXX Family of Devices

	Clock	Memory					Peripherals						Features			
	Maximum Frequency of Operation (MHz)	EPROM	ROM	Program Memory (Words)	RAM Data Memory (bytes)	Timer Module(s)	Captures	PWMs	Serial Port(s) (USART)	Hardware Multiply	External Interrupts	Interrupt Sources	I/O Pins	Voltage Range (Volts)	Number of Instructions	Packages
PIC17C42	25	2K	—	232		TMR0,TMR1, TMR2,TMR3	2	2	Yes	—	Yes	11	33	4.5-5.5	55	40-pin DIP; 44-pin PLCC, MQFP
PIC17C42A	25	2K	—	232		TMR0,TMR1, TMR2,TMR3	2	2	Yes	Yes	Yes	11	33	2.5-5.5	58	40-pin DIP; 44-pin PLCC, MQFP
PIC17CR42	25	—	2K	232		TMR0,TMR1, TMR2,TMR3	2	2	Yes	Yes	Yes	11	33	2.5-5.5	58	40-pin DIP; 44-pin PLCC, MQFP
PIC17C43	25	4K	—	454		TMR0,TMR1, TMR2,TMR3	2	2	Yes	Yes	Yes	11	33	2.5-6.0	58	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17CR43	25	—	4K	454		TMR0,TMR1, TMR2,TMR3	2	2	Yes	Yes	Yes	11	33	2.5-6.0	58	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17C44	25	8K		454		TMR0,TMR1, TMR2,TMR3	2	2	Yes	Yes	Yes	11	33	2.5-6.0	58	40-pin DIP; 44-pin PLCC, TQFP, MQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.

PIC14000 PRODUCT IDENTIFICATION SYSTEM

To order or to obtain information (e.g., on pricing or delivery), please use the listed part numbers, and refer to the factory or the listed sales offices.