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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	33MHz
Connectivity	UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	33
Program Memory Size	4KB (2K x 16)
Program Memory Type	EPROM, UV
EEPROM Size	-
RAM Size	232 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-CDIP (0.600", 15.24mm) Window
Supplier Device Package	40-Cerdip
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c42a-jw

6.4.1 INDIRECT ADDRESSING REGISTERS

The PIC17C4X has four registers for indirect addressing. These registers are:

- INDF0 and FSR0
- INDF1 and FSR1

Registers INDF0 and INDF1 are not physically implemented. Reading or writing to these registers activates indirect addressing, with the value in the corresponding FSR register being the address of the data. The FSR is an 8-bit register and allows addressing anywhere in the 256-byte data memory address range. For banked memory, the bank of memory accessed is specified by the value in the BSR.

If file INDF0 (or INDF1) itself is read indirectly via an FSR, all '0's are read (Zero bit is set). Similarly, if INDF0 (or INDF1) is written to indirectly, the operation will be equivalent to a NOP, and the status bits are not affected.

6.4.2 INDIRECT ADDRESSING OPERATION

The indirect addressing capability has been enhanced over that of the PIC16CXX family. There are two control bits associated with each FSR register. These two bits configure the FSR register to:

- Auto-decrement the value (address) in the FSR after an indirect access
- Auto-increment the value (address) in the FSR after an indirect access
- No change to the value (address) in the FSR after an indirect access

These control bits are located in the ALUSTA register. The FSR1 register is controlled by the FS3:FS2 bits and FSR0 is controlled by the FS1:FS0 bits.

When using the auto-increment or auto-decrement features, the effect on the FSR is not reflected in the ALUSTA register. For example, if the indirect address causes the FSR to equal '0', the Z bit will not be set.

If the FSR register contains a value of 0h, an indirect read will read 0h (Zero bit is set) while an indirect write will be equivalent to a NOP (status bits are not affected).

Indirect addressing allows single cycle data transfers within the entire data space. This is possible with the use of the `MOVPF` and `MOVFP` instructions, where either 'p' or 'f' is specified as INDF0 (or INDF1).

If the source or destination of the indirect address is in banked memory, the location accessed will be determined by the value in the BSR.

A simple program to clear RAM from 20h - FFh is shown in Example 6-1.

EXAMPLE 6-1: INDIRECT ADDRESSING

```
    MOVLW    0x20          ;
    MOVWF    FSR0          ; FSR0 = 20h
    BCF      ALUSTA, FS1   ; Increment FSR
    BSF      ALUSTA, FS0   ; after access
    BCF      ALUSTA, C     ; C = 0
    MOVLW    END_RAM + 1   ;
LP  CLRF     INDF0         ; Addr(FSR) = 0
    CPFSEQ   FSR0          ; FSR0 = END_RAM+1?
    GOTO     LP            ; NO, clear next
    :                      ; YES, All RAM is
    :                      ; cleared
```

6.5 Table Pointer (TBLPTRL and TBLPTRH)

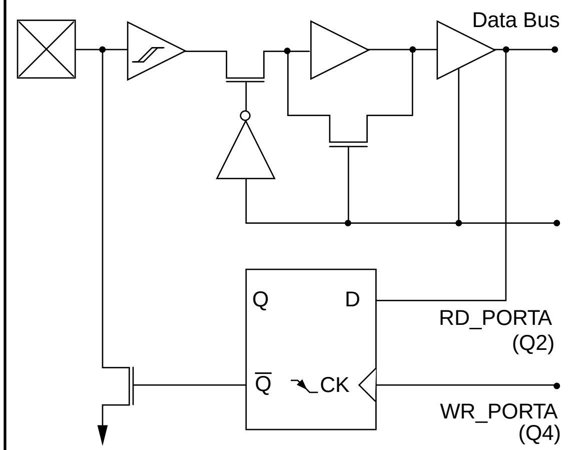
File registers TBLPTRL and TBLPTRH form a 16-bit pointer to address the 64K program memory space. The table pointer is used by instructions `TABLWT` and `TABLRD`.

The `TABLRD` and the `TABLWT` instructions allow transfer of data between program and data space. The table pointer serves as the 16-bit address of the data word within the program memory. For a more complete description of these registers and the operation of Table Reads and Table Writes, see Section 7.0.

6.6 Table Latch (TBLATH, TBLATL)

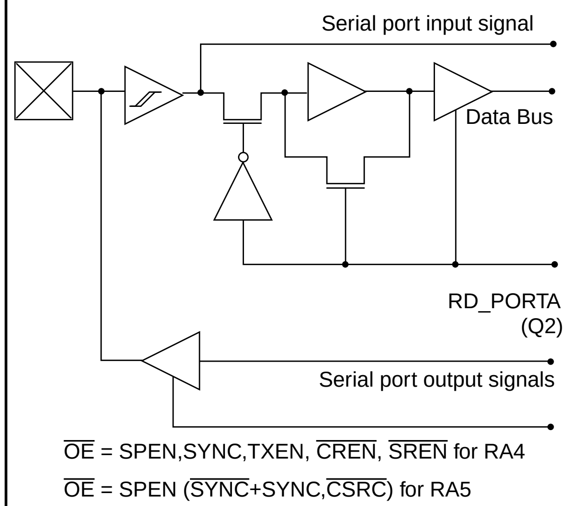
The table latch (TBLAT) is a 16-bit register, with TBLATH and TBLATL referring to the high and low bytes of the register. It is not mapped into data or program memory. The table latch is used as a temporary holding latch during data transfer between program and data memory (see descriptions of instructions `TABLRD`, `TABLWT`, `TLRD` and `TLWT`). For a more complete description of these registers and the operation of Table Reads and Table Writes, see Section 7.0.

FIGURE 9-2: RA2 AND RA3 BLOCK DIAGRAM



Note: I/O pins have protection diodes to VSS.

FIGURE 9-3: RA4 AND RA5 BLOCK DIAGRAM



Note: I/O pins have protection diodes to VDD and VSS.

TABLE 9-1: PORTA FUNCTIONS

Name	Bit0	Buffer Type	Function
RA0/INT	bit0	ST	Input or external interrupt input.
RA1/T0CKI	bit1	ST	Input or clock input to the TMR0 timer/counter, and/or an external interrupt input.
RA2	bit2	ST	Input/Output. Output is open drain type.
RA3	bit3	ST	Input/Output. Output is open drain type.
RA4/RX/DT	bit4	ST	Input or USART Asynchronous Receive or USART Synchronous Data.
RA5/TX/CK	bit5	ST	Input or USART Asynchronous Transmit or USART Synchronous Clock.
$\overline{RBPU}$	bit7	—	Control bit for PORTB weak pull-ups.

Legend: ST = Schmitt Trigger input.

TABLE 9-2: REGISTERS/BITS ASSOCIATED WITH PORTA

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other resets (Note1)
10h, Bank 0	PORTA	$\overline{RBPU}$	—	RA5	RA4	RA3	RA2	RA1/T0CKI	RA0/INT	0-xx xxxx	0-uu uuuu
05h, Unbanked	T0STA	INTEDG	T0SE	T0CS	PS3	PS2	PS1	PS0	—	0000 000-	0000 000-
13h, Bank 0	RCSTA	SPEN	RC9	SREN	CREN	—	FERR	OERR	RC9D	0000 -00x	0000 -00u
15h, Bank 0	TXSTA	CSRC	TX9	TXEN	SYNC	—	—	TRMT	TX9D	0000 --1x	0000 --1u

Legend: x = unknown, u = unchanged, - = unimplemented reads as '0'. Shaded cells are not used by PORTA.
Note 1: Other (non power-up) resets include: external reset through $\overline{MCLR}$ and the Watchdog Timer Reset.

9.4.1 PORTE AND DDRE REGISTER

PORTE is a 3-bit bi-directional port. The corresponding data direction register is DDRE. A '1' in DDRE configures the corresponding port pin as an input. A '0' in the DDRE register configures the corresponding port pin as an output. Reading PORTE reads the status of the pins, whereas writing to it will write to the port latch. PORTE is multiplexed with the system bus. When operating as the system bus, PORTE contains the control signals for the address/data bus (AD15:AD0). These control signals are Address Latch Enable (ALE), Output Enable ($\overline{OE}$), and Write ($\overline{WR}$). The control signals $\overline{OE}$ and $\overline{WR}$ are active low signals. The timing for the system bus is shown in the Electrical Characteristics section.

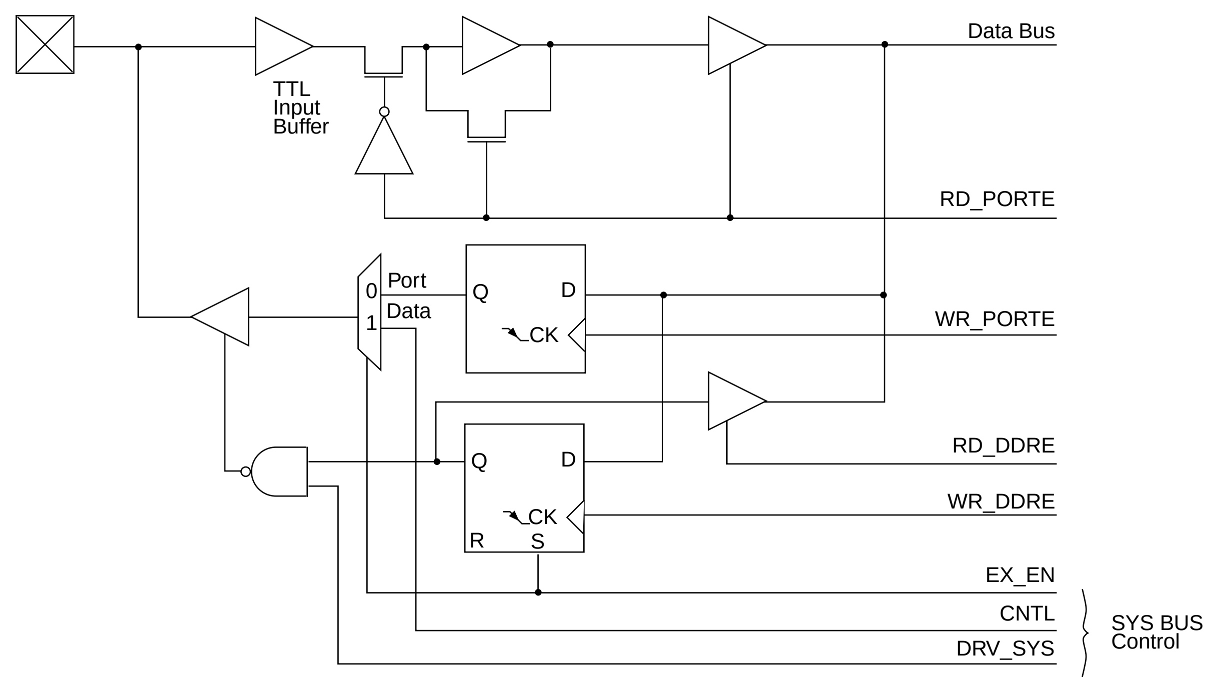
> **Note:** This port is configured as the system bus when the device's configuration bits are selected to Microprocessor or Extended Microcontroller modes. In the two other microcontroller modes, this port is a general purpose I/O.

Example 9-4 shows the instruction sequence to initialize PORTE. The Bank Select Register (BSR) must be selected to Bank 1 for the port to be initialized.

EXAMPLE 9-4: INITIALIZING PORTE

```
MOVLB  1          ;  Select Bank 1
CLRF   PORTE      ;  Initialize PORTE data
                  ;   latches before setting
                  ;   the data direction
                  ;   register
MOVLW  0x03       ;  Value used to initialize
                  ;   data direction
MOVWF  DDRE       ;  Set RE<1:0> as inputs
                  ;   RE<2> as outputs
                  ;   RE<7:3> are always
                  ;   read as '0'
```

FIGURE 9-8: PORTE BLOCK DIAGRAM (IN I/O PORT MODE)



Note: I/O pins have protection diodes to VDD and Vss.

12.1.3.1 PWM PERIODS

The period of the PWM1 output is determined by Timer1 and its period register (PR1). The period of the PWM2 output can be software configured to use either Timer1 or Timer2 as the time-base. When TM2PW2 bit (PW2DCL<5>) is clear, the time-base is determined by TMR1 and PR1. When TM2PW2 is set, the time-base is determined by Timer2 and PR2.

Running two different PWM outputs on two different timers allows different PWM periods. Running both PWMs from Timer1 allows the best use of resources by freeing Timer2 to operate as an 8-bit timer. Timer1 and Timer2 can not be used as a 16-bit timer if either PWM is being used.

The PWM periods can be calculated as follows:

period of PWM1 =[(PR1) + 1] x 4TOSC

period of PWM2 =[(PR1) + 1] x 4TOSC or
[(PR2) + 1] x 4TOSC

The duty cycle of PWMx is determined by the 10-bit value DCx<9:0>. The upper 8-bits are from register PWxDCH and the lower 2-bits are from PWxDCL<7:6> (PWxDCH:PWxDCL<7:6>). Table 12-3 shows the maximum PWM frequency (FPWM) given the value in the period register.

The number of bits of resolution that the PWM can achieve depends on the operation frequency of the device as well as the PWM frequency (FPWM).

Maximum PWM resolution (bits) for a given PWM frequency:

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

The PWMx duty cycle is as follows:

PWMx Duty Cycle = (DCx) x TOSC

where DCx represents the 10-bit value from PWxDCH:PWxDCL.

If DCx = 0, then the duty cycle is zero. If PRx = PWxDCH, then the PWM output will be low for one to four Q-clock (depending on the state of the PWxDCL<7:6> bits). For a Duty Cycle to be 100%, the PWxDCH value must be greater then the PRx value.

The duty cycle registers for both PWM outputs are double buffered. When the user writes to these registers, they are stored in master latches. When TMR1 (or TMR2) overflows and a new PWM period begins, the master latch values are transferred to the slave latches and the PWMx pin is forced high.

Note: For PW1DCH, PW1DCL, PW2DCH and PW2DCL registers, a write operation writes to the "master latches" while a read operation reads the "slave latches". As a result, the user may not read back what was just written to the duty cycle registers.

The user should also avoid any "read-modify-write" operations on the duty cycle registers, such as: `ADDWF PW1DCH`. This may cause duty cycle outputs that are unpredictable.

TABLE 12-3: PWM FREQUENCY vs. RESOLUTION AT 25 MHz

PWM Frequency	Frequency (kHz)				
	24.4	**48.8**	**65.104**	**97.66**	**390.6**
PRx Value	0xFF	0x7F	0x5F	0x3F	0x0F
High Resolution	10-bit	9-bit	8.5-bit	8-bit	6-bit
Standard Resolution	8-bit	7-bit	6.5-bit	6-bit	4-bit

12.1.3.2 PWM INTERRUPTS

The PWM module makes use of TMR1 or TMR2 interrupts. A timer interrupt is generated when TMR1 or TMR2 equals its period register and is cleared to zero. This interrupt also marks the beginning of a PWM cycle. The user can write new duty cycle values before the timer roll-over. The TMR1 interrupt is latched into the TMR1IF bit and the TMR2 interrupt is latched into the TMR2IF bit. These flags must be cleared in software.

12.1.3.3 EXTERNAL CLOCK SOURCE

The PWMs will operate regardless of the clock source of the timer. The use of an external clock has ramifications that must be understood. Because the external TCLK12 input is synchronized internally (sampled once per instruction cycle), the time TCLK12 changes to the time the timer increments will vary by as much as TCY (one instruction cycle). This will cause jitter in the duty cycle as well as the period of the PWM output.

This jitter will be ±TCY, unless the external clock is synchronized with the processor clock. Use of one of the PWM outputs as the clock source to the TCLKx input, will supply a synchronized clock.

In general, when using an external clock source for PWM, its frequency should be much less than the device frequency (Fosc).

12.2.3 EXTERNAL CLOCK INPUT FOR TIMER3

When TMR3CS is set, the 16-bit TMR3 increments on the falling edge of clock input TCLK3. The input on the RB5/TCLK3 pin is sampled and synchronized by the internal phase clocks twice every instruction cycle. This causes a delay from the time a falling edge appears on TCLK3 to the time TMR3 is actually incremented. For the external clock input timing requirements, see the Electrical Specification section. Figure 12-9 shows the timing diagram when operating from an external clock.

12.2.4 READING/WRITING TIMER3

Since Timer3 is a 16-bit timer and only 8-bits at a time can be read or written, care should be taken when reading or writing while the timer is running. The best method to read or write the timer is to stop the timer, perform any read or write operation, and then restart Timer3 (using the TMR3ON bit). However, if it is necessary to keep Timer3 free-running, care must be taken. For writing to the 16-bit TMR3, Example 12-2 may be used. For reading the 16-bit TMR3, Example 12-3 may be used. Interrupts must be disabled during this routine.

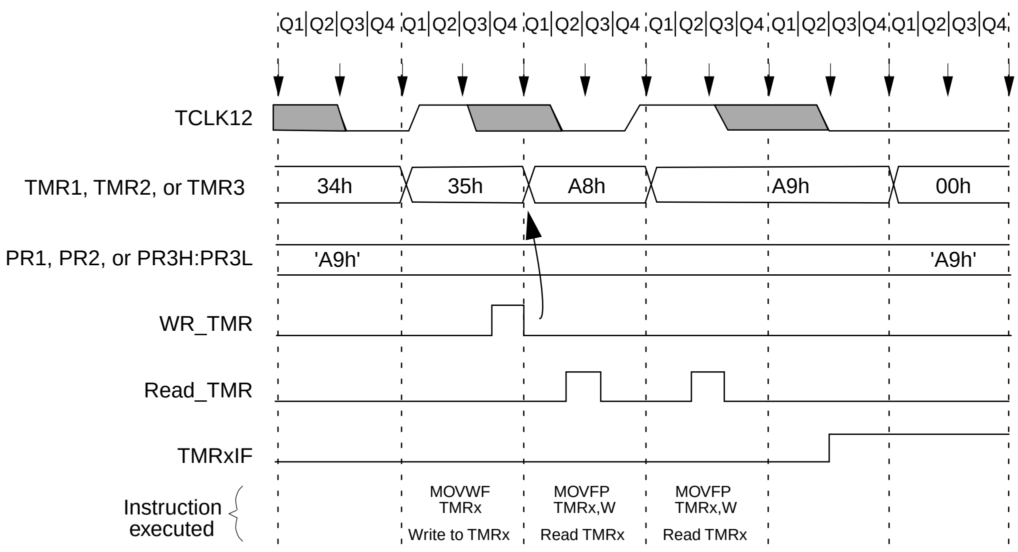
EXAMPLE 12-2: WRITING TO TMR3

```
BSF   CPUSTA, GLINTD ;Disable interrupt
MOVFP RAM_L,  TMR3L  ;
MOVFP RAM_H,  TMR3H  ;
BCF   CPUSTA, GLINTD ;Done,enable interrupt
```

EXAMPLE 12-3: READING FROM TMR3

```
MOVPF   TMR3L, TMPLO    ;read low tmr0
MOVPF   TMR3H, TMPHI    ;read high tmr0
MOVFP   TMPLO, WREG     ;tmplo —> wreg
CPFSLT  TMR3L, WREG     ;tmr0l < wreg?
RETURN                  ;no then return
MOVPF   TMR3L, TMPLO    ;read low tmr0
MOVPF   TMR3H, TMPHI    ;read high tmr0
RETURN                  ;return
```

FIGURE 12-9: TMR1, TMR2, AND TMR3 OPERATION IN EXTERNAL CLOCK MODE



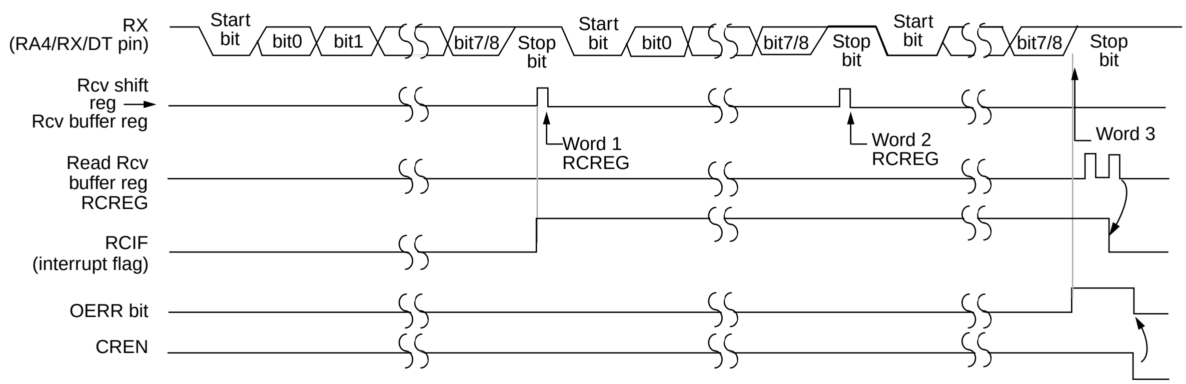
Note 1: TCLK12 is sampled in Q2 and Q4.
2: ↓ indicates a sampling point.
3: The latency from TCLK12 ↓ to timer increment is between 2Tosc and 6Tosc.

Steps to follow when setting up an Asynchronous Reception:

1. Initialize the SPBRG register for the appropriate baud rate.
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If interrupts are desired, then set the RCIE bit.
4. If 9-bit reception is desired, then set the RX9 bit.
5. Enable the reception by setting the CREN bit.
6. The RCIF bit will be set when reception completes and an interrupt will be generated if the RCIE bit was set.
7. Read RCSTA to get the ninth bit (if enabled) and FERR bit to determine if any error occurred during reception.
8. Read RCREG for the 8-bit received data.
9. If an overrun error occurred, clear the error by clearing the OERR bit.

Note:	To terminate a reception, either clear the SREN and CREN bits, or the SPEN bit. This will reset the receive logic, so that it will be in the proper state when receive is re-enabled.

FIGURE 13-8: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

TABLE 13-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other resets (Note1)
16h, Bank 1	PIR	RBIF	TMR3IF	TMR2IF	TMR1IF	CA2IF	CA1IF	TXIF	RCIF	0000 0010	0000 0010
13h, Bank 0	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00u
14h, Bank 0	RCREG	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0	xxxx xxxx	uuuu uuuu
17h, Bank 1	PIE	RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TXIE	RCIE	0000 0000	0000 0000
15h, Bank 0	TXSTA	CSRC	TX9	TXEN	SYNC	—	—	TRMT	TX9D	0000 --1x	0000 --1u
17h, Bank 0	SPBRG	Baud rate generator register								xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented read as a '0', shaded cells are not used for asynchronous reception.
Note 1: Other (non power-up) resets include: external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

ANDWF — AND WREG with f

Syntax: [*label*] ANDWF f,d

Operands: $0 \le f \le 255$
$d \in [0,1]$

Operation: (WREG) .AND. (f) → (dest)

Status Affected: Z

Encoding:

0000	101d	ffff	ffff

Description: The contents of WREG are AND'ed with register 'f'. If 'd' is 0 the result is stored in WREG. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	Write to destination

Example: `ANDWF REG, 1`

Before Instruction
WREG = 0x17
REG = 0xC2

After Instruction
WREG = 0x17
REG = 0x02

BCF — Bit Clear f

Syntax: [*label*] BCF f,b

Operands: $0 \le f \le 255$
$0 \le b \le 7$

Operation: 0 → (f<b>)

Status Affected: None

Encoding:

1000	1bbb	ffff	ffff

Description: Bit 'b' in register 'f' is cleared.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	Write register 'f'

Example: `BCF FLAG_REG, 7`

Before Instruction
FLAG_REG = 0xC7

After Instruction
FLAG_REG = 0x47

CPFSEQ — Compare f with WREG, skip if f = WREG

Syntax:	[*label*] CPFSEQ f
Operands:	0 ≤ f ≤ 255
Operation:	(f) – (WREG), skip if (f) = (WREG) (unsigned comparison)
Status Affected:	None
Encoding:	`0011` `0001` `ffff` `ffff`
Description:	Compares the contents of data memory location 'f' to the contents of WREG by performing an unsigned subtraction. If 'f' = WREG then the fetched instruction is discarded and an NOP is executed instead making this a two-cycle instruction.
Words:	1
Cycles:	1 (2)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	NOP

If skip:

Q1	Q2	Q3	Q4
Forced NOP	NOP	Execute	NOP

Example:

```
HERE     CPFSEQ REG
NEQUAL   :
EQUAL    :
```

Before Instruction
PC Address = HERE
WREG = ?
REG = ?

After Instruction
If REG = WREG;
PC = Address (EQUAL)
If REG ≠ WREG;
PC = Address (NEQUAL)

CPFSGT — Compare f with WREG, skip if f > WREG

Syntax:	[*label*] CPFSGT f
Operands:	0 ≤ f ≤ 255
Operation:	(f) – (WREG), skip if (f) > (WREG) (unsigned comparison)
Status Affected:	None
Encoding:	`0011` `0010` `ffff` `ffff`
Description:	Compares the contents of data memory location 'f' to the contents of the WREG by performing an unsigned subtraction. If the contents of 'f' > the contents of WREG then the fetched instruction is discarded and an NOP is executed instead making this a two-cycle instruction.
Words:	1
Cycles:	1 (2)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	NOP

If skip:

Q1	Q2	Q3	Q4
Forced NOP	NOP	Execute	NOP

Example:

```
HERE       CPFSGT REG
NGREATER   :
GREATER    :
```

Before Instruction
PC = Address (HERE)
WREG = ?

After Instruction
If REG > WREG;
PC = Address (GREATER)
If REG ≤ WREG;
PC = Address (NGREATER)

NEGW — Negate W

Syntax:	[*label*] NEGW f,s
Operands:	$0 \le F \le 255$ $s \in [0,1]$
Operation:	$\overline{WREG} + 1 \rightarrow (f)$; $\overline{WREG} + 1 \rightarrow s$
Status Affected:	OV, C, DC, Z
Encoding:	`0010` `110s` `ffff` `ffff`
Description:	WREG is negated using two's complement. If 's' is 0 the result is placed in WREG and data memory location 'f'. If 's' is 1 the result is placed only in data memory location 'f'.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	Write register 'f' and other specified register

Example: `NEGW REG,0`

Before Instruction
WREG = `0011 1010` [0x3A],
REG = `1010 1011` [0xAB]

After Instruction
WREG = `1100 0111` [0xC6]
REG = `1100 0111` [0xC6]

NOP — No Operation

Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Encoding:	`0000` `0000` `0000` `0000`
Description:	No operation.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	NOP	Execute	NOP

Example:

None.

SLEEP — Enter SLEEP mode

Syntax:	[*label*] SLEEP
Operands:	None
Operation:	00h → WDT; 0 → WDT postscaler; 1 → $\overline{TO}$; 0 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Encoding:	`0000 0000 0000 0011`
Description:	The power down status bit ($\overline{PD}$) is cleared. The time-out status bit ($\overline{TO}$) is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register PCLATH	Execute	NOP

Example: `SLEEP`

Before Instruction
$\overline{TO}$ = ?
$\overline{PD}$ = ?

After Instruction
$\overline{TO}$ = 1 †
$\overline{PD}$ = 0

† If WDT causes wake-up, this bit is cleared

SUBLW — Subtract WREG from Literal

Syntax:	[*label*] SUBLW k
Operands:	0 ≤ k ≤ 255
Operation:	k – (WREG) → (WREG)
Status Affected:	OV, C, DC, Z
Encoding:	`1011 0010 kkkk kkkk`
Description:	WREG is subtracted from the eight bit literal 'k'. The result is placed in WREG.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Execute	Write to WREG

Example 1: `SUBLW 0x02`

Before Instruction
WREG = 1
C = ?

After Instruction
WREG = 1
C = 1 ; result is positive
Z = 0

Example 2:

Before Instruction
WREG = 2
C = ?

After Instruction
WREG = 0
C = 1 ; result is zero
Z = 1

Example 3:

Before Instruction
WREG = 3
C = ?

After Instruction
WREG = FF ; (2's complement)
C = 0 ; result is negative
Z = 1

16.0 DEVELOPMENT SUPPORT

16.1 Development Tools

The PIC16/17 microcontrollers are supported with a full range of hardware and software development tools:

- PICMASTER/PICMASTER CE Real-Time In-Circuit Emulator
- ICEPIC Low-Cost PIC16C5X and PIC16CXXX In-Circuit Emulator
- PRO MATE® II Universal Programmer
- PICSTART® Plus Entry-Level Prototype Programmer
- PICDEM-1 Low-Cost Demonstration Board
- PICDEM-2 Low-Cost Demonstration Board
- PICDEM-3 Low-Cost Demonstration Board
- MPASM Assembler
- MPLAB-SIM Software Simulator
- MPLAB-C (C Compiler)
- Fuzzy logic development system (fuzzyTECH®–MP)

16.2 PICMASTER: High Performance Universal In-Circuit Emulator with MPLAB IDE

The PICMASTER Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for all microcontrollers in the PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX and PIC17CXX families. PICMASTER is supplied with the MPLAB™ Integrated Development Environment (IDE), which allows editing, "make" and download, and source debugging from a single environment.

Interchangeable target probes allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the PICMASTER allows expansion to support all new Microchip microcontrollers.

The PICMASTER Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC compatible 386 (and higher) machine platform and Microsoft Windows® 3.x environment were chosen to best make these features available to you, the end user.

A CE compliant version of PICMASTER is available for European Union (EU) countries.

16.3 ICEPIC: Low-cost PIC16CXXX In-Circuit Emulator

ICEPIC is a low-cost in-circuit emulator solution for the Microchip PIC16C5X and PIC16CXXX families of 8-bit OTP microcontrollers.

ICEPIC is designed to operate on PC-compatible machines ranging from 286-AT® through Pentium™ based machines under Windows 3.x environment. ICEPIC features real time, non-intrusive emulation.

16.4 PRO MATE II: Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode.

The PRO MATE II has programmable V_{DD} and V_{PP} supplies which allows it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for displaying error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PIC16C5X, PIC16CXXX, PIC17CXX and PIC14000 devices. It can also set configuration and code-protect bits in this mode.

16.5 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. PICSTART Plus is not recommended for production programming.

PICSTART Plus supports all PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX and PIC17CXX devices with up to 40 pins. Larger pin count devices such as the PIC16C923 and PIC16C924 may be supported with an adapter socket.

Applicable Devices	42	R42	42A	43	R43	44

FIGURE 17-9: USART MODULE: SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

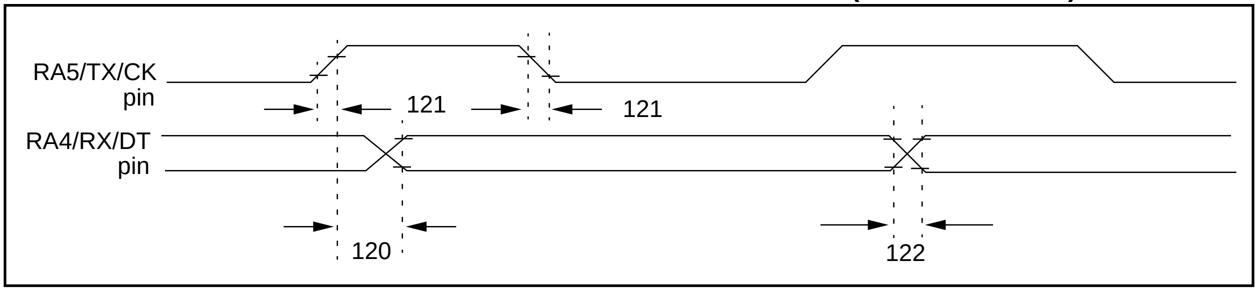


TABLE 17-9: SERIAL PORT SYNCHRONOUS TRANSMISSION REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE) Clock high to data out valid	—	—	65	ns	
121	TckRF	Clock out rise time and fall time (Master Mode)	—	10	35	ns	
122	TdtRF	Data out rise time and fall time	—	10	35	ns	

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

FIGURE 17-10: USART MODULE: SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING

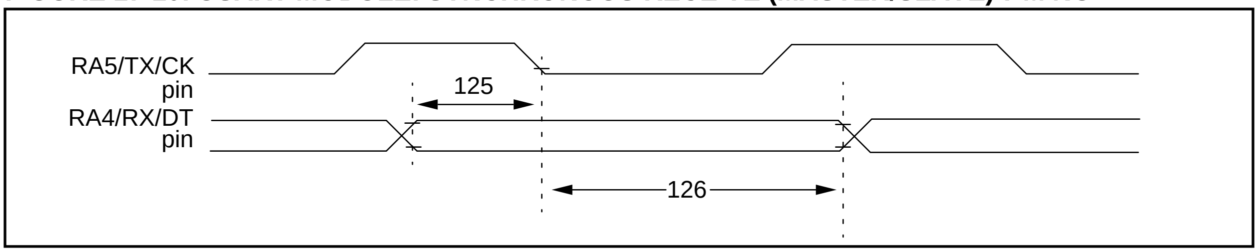


TABLE 17-10: SERIAL PORT SYNCHRONOUS RECEIVE REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
125	TdtV2ckL	SYNC RCV (MASTER & SLAVE) Data hold before CK↓ (DT hold time)	15	—	—	ns	
126	TckL2dtl	Data hold after CK↓ (DT hold time)	15	—	—	ns	

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

Applicable Devices	42	R42	42A	43	R43	44

FIGURE 18-2: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

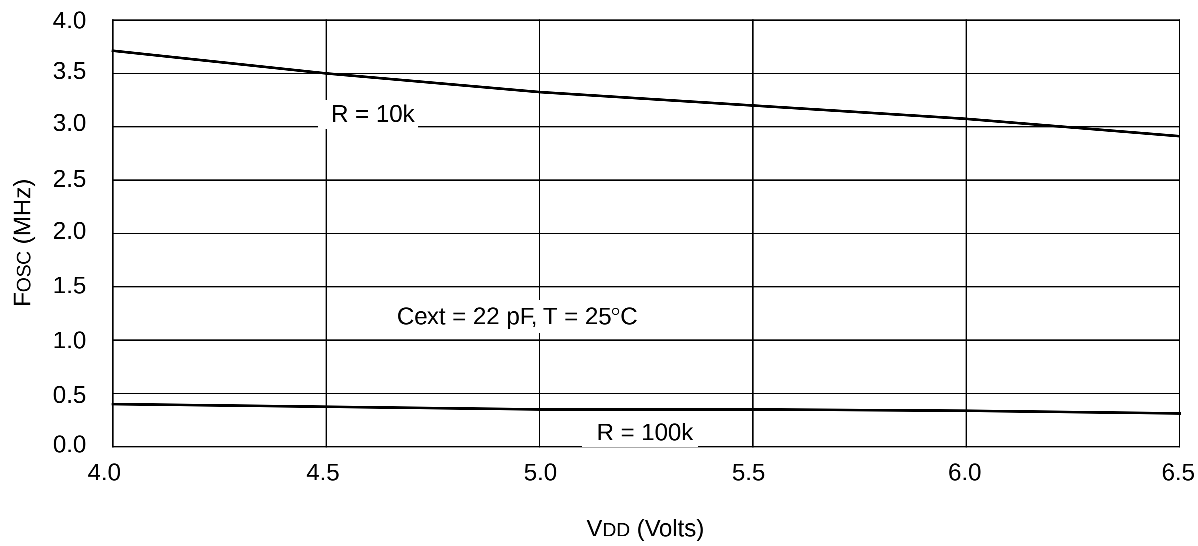
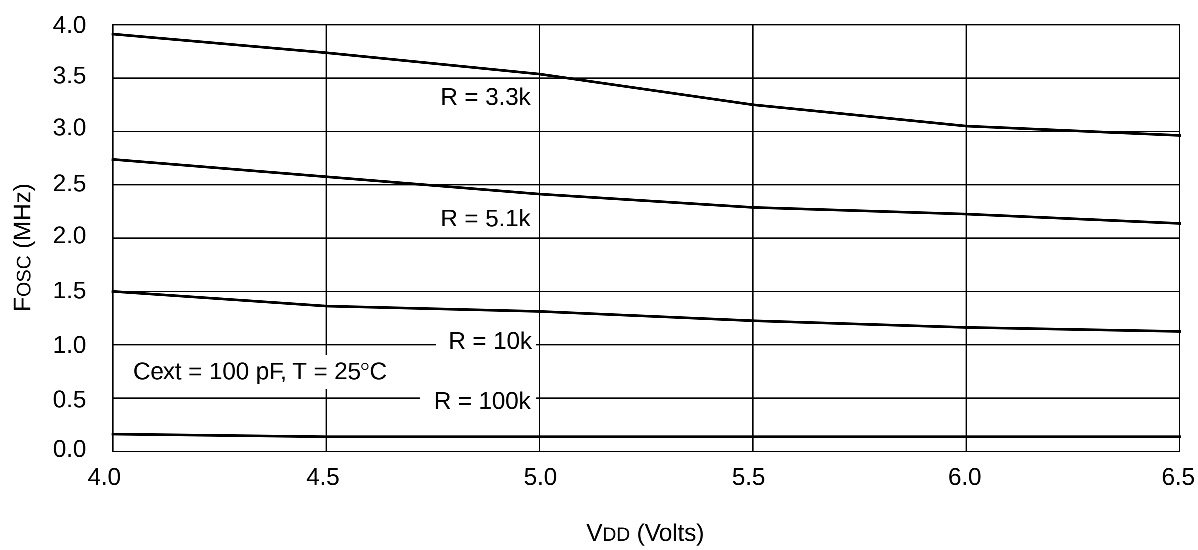


FIGURE 18-3: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD



Applicable Devices	42	R42	42A	43	R43	44

19.0 PIC17CR42/42A/43/R43/44 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

Ambient temperature under bias...-55 to +125°C
Storage temperature .... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} .... 0 to +7.5V
Voltage on $\overline{MCLR}$ with respect to V_{SS} (Note 2) ....-0.6V to +14V
Voltage on RA2 and RA3 with respect to V_{SS}.......................................-0.6V to +14V
Voltage on all other pins with respect to V_{SS} .... -0.6V to V_{DD} + 0.6V
Total power dissipation (Note 1)...1.0W
Maximum current out of V_{SS} pin(s) - total ....250 mA
Maximum current into V_{DD} pin(s) - total..200 mA
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{DD}$) ....±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{DD}$)......................................±20 mA
Maximum output current sunk by any I/O pin (except RA2 and RA3).........................35 mA
Maximum output current sunk by RA2 or RA3 pins ....60 mA
Maximum output current sourced by any I/O pin ....20 mA
Maximum current sunk by PORTA and PORTB (combined).......................................150 mA
Maximum current sourced by PORTA and PORTB (combined)....................................100 mA
Maximum current sunk by PORTC, PORTD and PORTE (combined)................................150 mA
Maximum current sourced by PORTC, PORTD and PORTE (combined).............................100 mA

Note 1: Power dissipation is calculated as follows: $Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD}-V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

Note 2: Voltage spikes below V_{SS} at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin rather than pulling this pin directly to V_{SS}.

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Applicable Devices	42	R42	42A	43	R43	44

19.3 DC CHARACTERISTICS: PIC17CR42/42A/43/R43/44-16 (Commercial, Industrial) PIC17CR42/42A/43/R43/44-25 (Commercial, Industrial) PIC17CR42/42A/43/R43/44-33 (Commercial, Industrial) PIC17LCR42/42A/43/R43/44-08 (Commercial, Industrial)

DC CHARACTERISTICS			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40˚C ≤ TA ≤ +85˚C for industrial and 0˚C ≤ TA ≤ +70˚C for commercial Operating voltage VDD range as described in Section 19.1				
Parameter No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
		Input Low Voltage					
	VIL	I/O ports					
D030		with TTL buffer	VSS	–	0.8	V	4.5V ≤ VDD ≤ 5.5V
			VSS	–	0.2VDD	V	2.5V ≤ VDD ≤ 4.5V
D031		with Schmitt Trigger buffer	VSS	–	0.2VDD	V	
D032		$\overline{MCLR}$, OSC1 (in EC and RC mode)	Vss	–	0.2VDD	V	Note1
D033		OSC1 (in XT, and LF mode)	–	0.5VDD	–	V	
		Input High Voltage					
	VIH	I/O ports					
D040		with TTL buffer	2.0	–	VDD	V	4.5V ≤ VDD ≤ 5.5V
			1 + 0.2VDD	–	VDD	V	2.5V ≤ VDD ≤ 4.5V
D041		with Schmitt Trigger buffer	0.8VDD	–	VDD	V	
D042		$\overline{MCLR}$	0.8VDD	–	VDD	V	Note1
D043		OSC1 (XT, and LF mode)	–	0.5VDD	–	V	
D050	VHYS	Hysteresis of Schmitt Trigger inputs	0.15VDD *	–	–	V	
		Input Leakage Current (Notes 2, 3)					
D060	IIL	I/O ports (except RA2, RA3)	–	–	±1	µA	Vss ≤ VPIN ≤ VDD, I/O Pin at hi-impedance PORTB weak pull-ups disabled
D061		$\overline{MCLR}$	–	–	±2	µA	VPIN = Vss or VPIN = VDD
D062		RA2, RA3			±2	µA	Vss ≤ VRA2, VRA3 ≤ 12V
D063		OSC1, TEST (EC, RC modes)	–	–	±1	µA	Vss ≤ VPIN ≤ VDD
D063B		OSC1, TEST (XT, LF modes)	–	–	VPIN	µA	RF ≥ 1 MΩ, see Figure 14.2
D064		$\overline{MCLR}$	–	–	10	µA	VMCLR = VPP = 12V (when not programming)
D070	IPURB	PORTB weak pull-up current	60	200	400	µA	VPIN = VSS, $\overline{RBPU}$ = 0 4.5V ≤ VDD ≤ 6.0V

* These parameters are characterized but not tested.

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

‡ These parameters are for design guidance only and are not tested, nor characterized.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC17CXX devices be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as coming out of the pin.

4: These specifications are for the programming of the on-chip program memory EPROM through the use of the table write instructions. The complete programming specifications can be found in: PIC17CXX Programming Specifications (Literature number DS30139).

5: The $\overline{MCLR}$/VPP pin may be kept in this range at times other than programming, but is not recommended.

6: For TTL buffers, the better of the two specifications may be used.

Applicable Devices	42	R42	42A	43	R43	44

FIGURE 19-11: MEMORY INTERFACE WRITE TIMING (NOT SUPPORTED IN PIC17LC4X DEVICES)

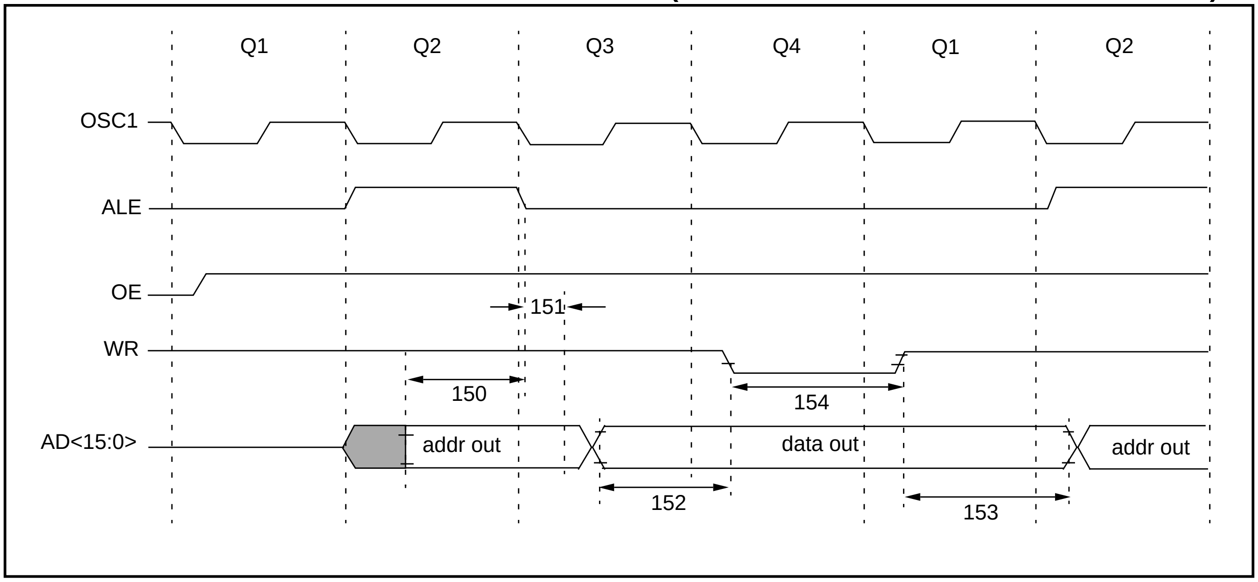


TABLE 19-11: MEMORY INTERFACE WRITE REQUIREMENTS (NOT SUPPORTED IN PIC17LC4X DEVICES)

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
150	TadV2alL	AD<15:0> (address) valid to ALE↓ (address setup time)	0.25Tcy - 10	—	—	ns	
151	TalL2adI	ALE↓ to address out invalid (address hold time)	0	—	—	ns	
152	TadV2wrL	Data out valid to $\overline{WR}$↓ (data setup time)	0.25Tcy - 40	—	—	ns	
153	TwrH2adI	$\overline{WR}$↑ to data out invalid (data hold time)	—	0.25TCY §	—	ns	
154	TwrL	$\overline{WR}$ pulse width	—	0.25TCY §	—	ns	

* These parameters are characterized but not tested.
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
§ This specification ensured by design.

Applicable Devices	42	R42	42A	43	R43	44

FIGURE 20-15: IOH vs. VOH, VDD = 5V

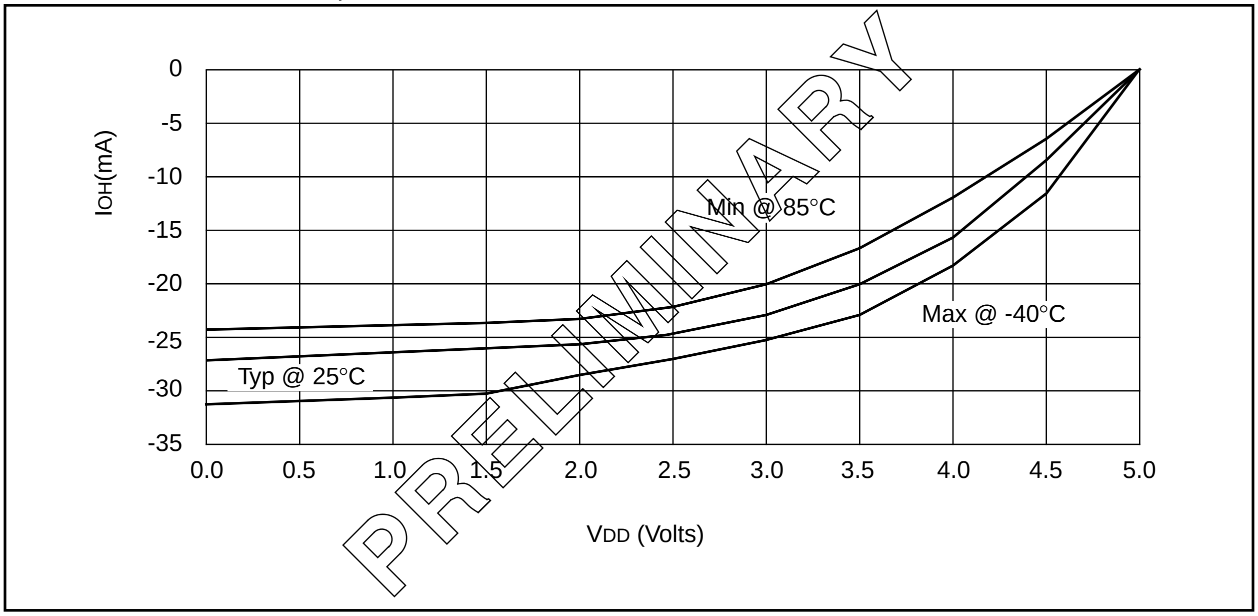
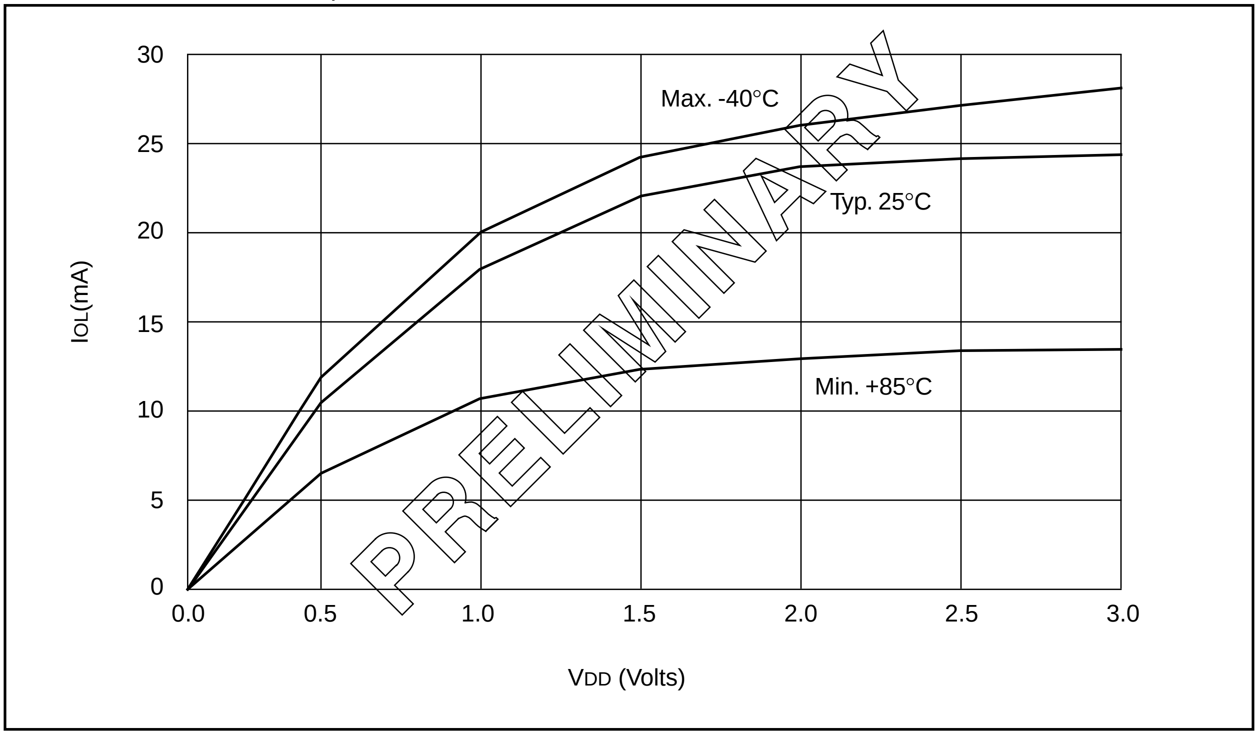
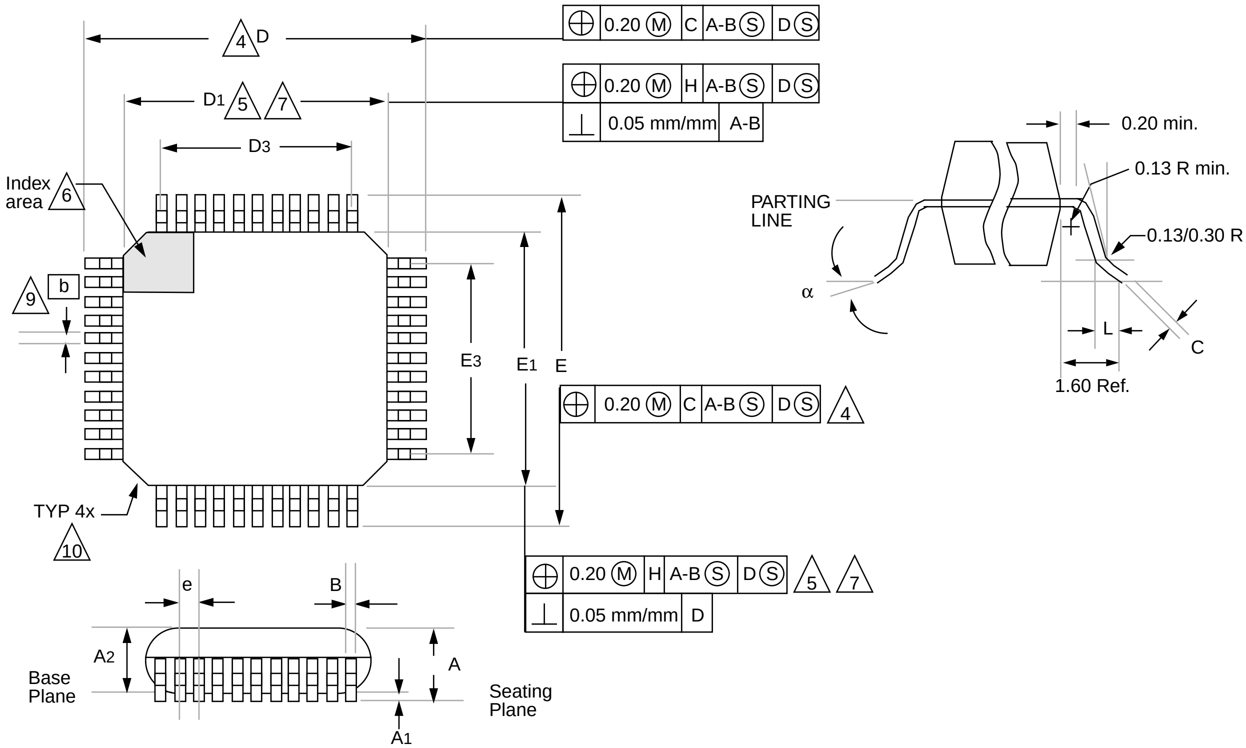


FIGURE 20-16: IOL vs. VOL, VDD = 3V



21.4 44-Lead Plastic Surface Mount (MQFP 10x10 mm Body 1.6/0.15 mm Lead Form)



Package Group: Plastic MQFP						
	Millimeters			Inches		
Symbol	Min	Max	Notes	Min	Max	Notes
α	0°	7°		0°	7°	
A	2.000	2.350		0.078	0.093	
A1	0.050	0.250		0.002	0.010	
A2	1.950	2.100		0.768	0.083	
b	0.300	0.450	**Typical**	0.011	0.018	**Typical**
C	0.150	0.180		0.006	0.007	
D	12.950	13.450		0.510	0.530	
D1	9.900	10.100		0.390	0.398	
D3	8.000	8.000	**Reference**	0.315	0.315	**Reference**
E	12.950	13.450		0.510	0.530	
E1	9.900	10.100		0.390	0.398	
E3	8.000	8.000	**Reference**	0.315	0.315	**Reference**
e	0.800	0.800		0.031	0.032	
L	0.730	1.030		0.028	0.041	
N	44	44		44	44	
CP	0.102	–		0.004	–	

E.2 PIC16C5X Family of Devices

	Clock	Memory			Peripherals	Features			
	Maximum Frequency of Operation (MHz)	Program Memory (x12 words) EPROM	Program Memory (x12 words) ROM	RAM Data Memory (bytes)	Timer Module(s)	I/O Pins	Voltage Range (Volts)	Number of Instructions	Packages
PIC16C52	4	384	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC
PIC16C54	20	512	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C54A	20	512	—	25	TMR0	12	2.0-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR54A	20	—	512	25	TMR0	12	2.0-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C55	20	512	—	24	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16C56	20	1K	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C57	20	2K	—	72	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16CR57B	20	—	2K	72	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16C58A	20	2K	—	73	TMR0	12	2.0-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR58A	20	—	2K	73	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP

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

Figure 6-12: Program Counter using The CALL and GOTO Instructions 41
Figure 6-13: BSR Operation (PIC17C43/R43/44) 42
Figure 7-1: TLWT Instruction Operation 43
Figure 7-2: TABLWT Instruction Operation 43
Figure 7-3: TLRD Instruction Operation 44
Figure 7-4: TABLRD Instruction Operation 44
Figure 7-5: TABLWT Write Timing (External Memory) 46
Figure 7-6: Consecutive TABLWT Write Timing (External Memory) 47
Figure 7-7: TABLRD Timing 48
Figure 7-8: TABLRD Timing (Consecutive TABLRD Instructions) 48
Figure 9-1: RA0 and RA1 Block Diagram 53
Figure 9-2: RA2 and RA3 Block Diagram 54
Figure 9-3: RA4 and RA5 Block Diagram 54
Figure 9-4: Block Diagram of RB<7:4> and RB<1:0> Port Pins 55
Figure 9-5: Block Diagram of RB3 and RB2 Port Pins 56
Figure 9-6: Block Diagram of RC<7:0> Port Pins 58
Figure 9-7: PORTD Block Diagram (in I/O Port Mode) 60
Figure 9-8: PORTE Block Diagram (in I/O Port Mode) 62
Figure 9-9: Successive I/O Operation 64
Figure 11-1: T0STA Register (Address: 05h, Unbanked) 67
Figure 11-2: Timer0 Module Block Diagram 68
Figure 11-3: TMR0 Timing with External Clock (Increment on Falling Edge) 68
Figure 11-4: TMR0 Timing: Write High or Low Byte 69
Figure 11-5: TMR0 Read/Write in Timer Mode 70
Figure 12-1: TCON1 Register (Address: 16h, Bank 3) 71
Figure 12-2: TCON2 Register (Address: 17h, Bank 3) 72
Figure 12-3: Timer1 and Timer2 in Two 8-bit Timer/Counter Mode 73
Figure 12-4: TMR1 and TMR2 in 16-bit Timer/Counter Mode 74
Figure 12-5: Simplified PWM Block Diagram 75
Figure 12-6: PWM Output 75
Figure 12-7: Timer3 with One Capture and One Period Register Block Diagram 78
Figure 12-8: Timer3 with Two Capture Registers Block Diagram 79
Figure 12-9: TMR1, TMR2, and TMR3 Operation in External Clock Mode 80
Figure 12-10: TMR1, TMR2, and TMR3 Operation in Timer Mode 81
Figure 13-1: TXSTA Register (Address: 15h, Bank 0) 83
Figure 13-2: RCSTA Register (Address: 13h, Bank 0) 84
Figure 13-3: USART Transmit 85
Figure 13-4: USART Receive 85
Figure 13-5: Asynchronous Master Transmission 90
Figure 13-6: Asynchronous Master Transmission (Back to Back) 90
Figure 13-7: RX Pin Sampling Scheme 91
Figure 13-8: Asynchronous Reception 92
Figure 13-9: Synchronous Transmission 94
Figure 13-10: Synchronous Transmission (Through TXEN) 94
Figure 13-11: Synchronous Reception (Master Mode, SREN) 95
Figure 14-1: Configuration Word 99
Figure 14-2: Crystal or Ceramic Resonator Operation (XT or LF OSC Configuration) 100
Figure 14-3: Crystal Operation, Overtone Crystals (XT OSC Configuration) 101
Figure 14-4: External Clock Input Operation (EC OSC Configuration) 101
Figure 14-5: External Parallel Resonant Crystal Oscillator Circuit 102
Figure 14-6: External Series Resonant Crystal Oscillator Circuit 102
Figure 14-7: RC Oscillator Mode 102
Figure 14-8: Watchdog Timer Block Diagram 104
Figure 14-9: Wake-up From Sleep Through Interrupt 105
Figure 15-1: General Format for Instructions 108
Figure 15-2: Q Cycle Activity 109
Figure 17-1: Parameter Measurement Information 154
Figure 17-2: External Clock Timing 155
Figure 17-3: CLKOUT and I/O Timing 156
Figure 17-4: Reset, Watchdog Timer, Oscillator Start-Up Timer and Power-Up Timer Timing 157
Figure 17-5: Timer0 Clock Timings 158
Figure 17-6: Timer1, Timer2, And Timer3 Clock Timings 158
Figure 17-7: Capture Timings 159
Figure 17-8: PWM Timings 159
Figure 17-9: USART Module: Synchronous Transmission (Master/Slave) Timing 160
Figure 17-10: USART Module: Synchronous Receive (Master/Slave) Timing 160
Figure 17-11: Memory Interface Write Timing 161
Figure 17-12: Memory Interface Read Timing 162
Figure 18-1: Typical RC Oscillator Frequency vs. Temperature 163
Figure 18-2: Typical RC Oscillator Frequency vs. V_{DD} 164
Figure 18-3: Typical RC Oscillator Frequency vs. V_{DD} 164
Figure 18-4: Typical RC Oscillator Frequency vs. V_{DD} 165
Figure 18-5: Transconductance (gm) of LF Oscillator vs. V_{DD} 166
Figure 18-6: Transconductance (gm) of XT Oscillator vs. V_{DD} 166
Figure 18-7: Typical I_{DD} vs. Frequency (External Clock 25°C) 167
Figure 18-8: Maximum I_{DD} vs. Frequency (External Clock 125°C to -40°C) 167
Figure 18-9: Typical I_{PD} vs. V_{DD} Watchdog Disabled 25°C 168
Figure 18-10: Maximum I_{PD} vs. V_{DD} Watchdog Disabled 168
Figure 18-11: Typical I_{PD} vs. V_{DD} Watchdog Enabled 25°C 169
Figure 18-12: Maximum I_{PD} vs. V_{DD} Watchdog Enabled 169
Figure 18-13: WDT Timer Time-Out Period vs. V_{DD} 170
Figure 18-14: I_{OH} vs. V_{OH}, V_{DD} = 3V 170
Figure 18-15: I_{OH} vs. V_{OH}, V_{DD} = 5V 171
Figure 18-16: I_{OL} vs. V_{OL}, V_{DD} = 3V 171
Figure 18-17: I_{OL} vs. V_{OL}, V_{DD} = 5V 172
Figure 18-18: V_{TH} (Input Threshold Voltage) of I/O Pins (TTL) vs. V_{DD} 172
Figure 18-19: V_{TH}, V_{IL} of I/O Pins (Schmitt Trigger) vs. V_{DD} 173
Figure 18-20: V_{TH} (Input Threshold Voltage) of OSC1 Input (In XT and LF Modes) vs. V_{DD} 173
Figure 19-1: Parameter Measurement Information 183

