



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	8KB (4K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	454 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-MQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c43-33e-pq

2.0 PIC17C4X DEVICE VARIETIES

A variety of frequency ranges and packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in the PIC17C4X Product Selection System section at the end of this data sheet. When placing orders, please use the "PIC17C4X Product Identification System" at the back of this data sheet to specify the correct part number.

For the PIC17C4X family of devices, there are four device "types" as indicated in the device number:

1. **C**, as in PIC17**C**42. These devices have EPROM type memory and operate over the standard voltage range.
2. **LC**, as in PIC17**LC**42. These devices have EPROM type memory, operate over an extended voltage range, and reduced frequency range.
3. **CR**, as in PIC17**CR**42. These devices have ROM type memory and operate over the standard voltage range.
4. **LCR**, as in PIC17**LCR**42. These devices have ROM type memory, operate over an extended voltage range, and reduced frequency range.

2.1 UV Erasable Devices

The UV erasable version, offered in Cerdip package, is optimal for prototype development and pilot programs.

The UV erasable version can be erased and reprogrammed to any of the configuration modes. Microchip's PRO MATE™ programmer supports programming of the PIC17C4X. Third party programmers also are available; refer to the *Third Party Guide* for a list of sources.

2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers expecting frequent code changes and updates.

The OTP devices, packaged in plastic packages, permit the user to program them once. In addition to the program memory, the configuration bits must also be programmed.

2.3 Quick-Turnaround-Production (QTP) Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who choose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices but with all EPROM locations and configuration options already programmed by the factory. Certain code and prototype verification procedures apply before production shipments are available. Please contact your local Microchip Technology sales office for more details.

2.4 Serialized Quick-Turnaround Production (SQTPSM) Devices

Microchip offers a unique programming service where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random or sequential.

Serial programming allows each device to have a unique number which can serve as an entry-code, password or ID number.

ROM devices do not allow serialization information in the program memory space.

For information on submitting ROM code, please contact your regional sales office.

2.5 Read Only Memory (ROM) Devices

Microchip offers masked ROM versions of several of the highest volume parts, thus giving customers a low cost option for high volume, mature products.

For information on submitting ROM code, please contact your regional sales office.

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC17C4X can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC17C4X uses a modified Harvard architecture. This architecture has the program and data accessed from separate memories. So the device has a program memory bus and a data memory bus. This improves bandwidth over traditional von Neumann architecture, where program and data are fetched from the same memory (accesses over the same bus). Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. PIC17C4X opcodes are 16-bits wide, enabling single word instructions. The full 16-bit wide program memory bus fetches a 16-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions execute in a single cycle (121 ns @ 33 MHz), except for program branches and two special instructions that transfer data between program and data memory.

The PIC17C4X can address up to 64K x 16 of program memory space.

The **PIC17C42** and **PIC17C42A** integrate 2K x 16 of EPROM program memory on-chip, while the **PIC17CR42** has 2K x 16 of ROM program memory on-chip.

The **PIC17C43** integrates 4K x 16 of EPROM program memory, while the **PIC17CR43** has 4K x 16 of ROM program memory.

The **PIC17C44** integrates 8K x 16 EPROM program memory.

Program execution can be internal only (microcontroller or protected microcontroller mode), external only (microprocessor mode) or both (extended microcontroller mode). Extended microcontroller mode does not allow code protection.

The PIC17CXX can directly or indirectly address its register files or data memory. All special function registers, including the Program Counter (PC) and Working Register (WREG), are mapped in the data memory. The PIC17CXX has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC17CXX simple yet efficient. In addition, the learning curve is reduced significantly.

One of the PIC17CXX family architectural enhancements from the PIC16CXX family allows two file registers to be used in some two operand instructions. This allows data to be moved directly between two registers without going through the WREG register. This increases performance and decreases program memory usage.

The PIC17CXX devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

The ALU is 8-bits wide and capable of addition, subtraction, shift, and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature.

The WREG register is an 8-bit working register used for ALU operations.

All PIC17C4X devices (except the PIC17C42) have an 8 x 8 hardware multiplier. This multiplier generates a 16-bit result in a single cycle.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow and digit borrow out bit, respectively, in subtraction. See the **SUBLW** and **SUBWF** instructions for examples.

Although the ALU does not perform signed arithmetic, the Overflow bit (OV) can be used to implement signed math. Signed arithmetic is comprised of a magnitude and a sign bit. The overflow bit indicates if the magnitude overflows and causes the sign bit to change state. Signed math can have greater than 7-bit values (magnitude), if more than one byte is used. The use of the overflow bit only operates on bit6 (MSb of magnitude) and bit7 (sign bit) of the value in the ALU. That is, the overflow bit is not useful if trying to implement signed math where the magnitude, for example, is 11-bits. If the signed math values are greater than 7-bits (15-, 24- or 31-bit), the algorithm must ensure that the low order bytes ignore the overflow status bit.

Care should be taken when adding and subtracting signed numbers to ensure that the correct operation is executed. Example 3-1 shows an item that must be taken into account when doing signed arithmetic on an ALU which operates as an unsigned machine.

EXAMPLE 3-1: SIGNED MATH

Hex Value	Signed Value Math	Unsigned Value Math
FFh	-127	255
+ 01h	+ 1	+ 1
= ?	= -126 (FEh)	= 0 (00h); Carry bit = 1

Signed math requires the result in REG to be FEh (-126). This would be accomplished by subtracting one as opposed to adding one.

Simplified block diagrams are shown in Figure 3-1 and Figure 3-2. The descriptions of the device pins are listed in Table 3-1.

4.1.3 OSCILLATOR START-UP TIMER (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle (1024Tosc) delay after $\overline{\text{MCLR}}$ is detected high or a wake-up from SLEEP event occurs.

The OST time-out is invoked only for XT and LF oscillator modes on a Power-on Reset or a Wake-up from SLEEP.

The OST counts the oscillator pulses on the OSC1/CLKIN pin. The counter only starts incrementing after the amplitude of the signal reaches the oscillator input thresholds. This delay allows the crystal oscillator or resonator to stabilize before the device exits reset. The length of time-out is a function of the crystal/resonator frequency.

4.1.4 TIME-OUT SEQUENCE

On power-up the time-out sequence is as follows: First the internal POR signal goes high when the POR trip point is reached. If $\overline{\text{MCLR}}$ is high, then both the OST and PWRT timers start. In general the PWRT time-out is longer, except with low frequency crystals/resonators. The total time-out also varies based on oscillator configuration. Table 4-1 shows the times that are associated with the oscillator configuration. Figure 4-2 and Figure 4-3 display these time-out sequences.

If the device voltage is not within electrical specification at the end of a time-out, the $\overline{\text{MCLR}}/\text{VPP}$ pin must be held low until the voltage is within the device specification. The use of an external RC delay is sufficient for many of these applications.

TABLE 4-1: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up	Wake up from SLEEP	MCLR Reset
XT, LF	Greater of: 96 ms or 1024Tosc	1024Tosc	—
EC, RC	Greater of: 96 ms or 1024Tosc	—	—

The time-out sequence begins from the first rising edge of $\overline{\text{MCLR}}$.

Table 4-3 shows the reset conditions for some special registers, while Table 4-4 shows the initialization conditions for all the registers. The shaded registers (in Table 4-4) are for all devices except the PIC17C42. In the PIC17C42, the PRODH and PRODL registers are general purpose RAM.

TABLE 4-2: STATUS BITS AND THEIR SIGNIFICANCE

$\overline{\text{TO}}$	$\overline{\text{PD}}$	Event
1	1	Power-on Reset, $\overline{\text{MCLR}}$ Reset during normal operation, or CLRWDT instruction executed
1	0	$\overline{\text{MCLR}}$ Reset during SLEEP or interrupt wake-up from SLEEP
0	1	WDT Reset during normal operation
0	0	WDT Reset during SLEEP

In Figure 4-2, Figure 4-3 and Figure 4-4, $\text{TPWRT} > \text{TOST}$, as would be the case in higher frequency crystals. For lower frequency crystals, (i.e., 32 kHz) TOST would be greater.

TABLE 4-3: RESET CONDITION FOR THE PROGRAM COUNTER AND THE CPUSTA REGISTER

Event		PCH:PCL	CPUSTA	OST Active
Power-on Reset		0000h	--11 11--	Yes
$\overline{\text{MCLR}}$ Reset during normal operation		0000h	--11 11--	No
$\overline{\text{MCLR}}$ Reset during SLEEP		0000h	--11 10--	Yes ⁽²⁾
WDT Reset during normal operation		0000h	--11 01--	No
WDT Reset during SLEEP ⁽³⁾		0000h	--11 00--	Yes ⁽²⁾
Interrupt wake-up from SLEEP	GLINTD is set	PC + 1	--11 10--	Yes ⁽²⁾
	GLINTD is clear	PC + 1 ⁽¹⁾	--10 10--	Yes ⁽²⁾

Legend: u = unchanged, x = unknown, - = unimplemented read as '0'.

Note 1: On wake-up, this instruction is executed. The instruction at the appropriate interrupt vector is fetched and then executed.

2: The OST is only active when the Oscillator is configured for XT or LF modes.

3: The Program Counter = 0, that is the device branches to the reset vector. This is different from the mid-range devices.

PIC17C4X

TABLE 4-4: INITIALIZATION CONDITIONS FOR SPECIAL FUNCTION REGISTERS (Cont.'d)

Register	Address	Power-on Reset	MCLR Reset WDT Reset	Wake-up from SLEEP through interrupt
Bank 2				
TMR1	10h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR2	11h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR3L	12h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR3H	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PR1	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PR2	15h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PR3/CA1L	16h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PR3/CA1H	17h	xxxx xxxx	uuuu uuuu	uuuu uuuu
Bank 3				
PW1DCL	10h	xx-- ----	uu-- ----	uu-- ----
PW2DCL	11h	xx-- ----	uu-- ----	uu-- ----
PW1DCH	12h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PW2DCH	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CA2L	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CA2H	15h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TCON1	16h	0000 0000	0000 0000	uuuu uuuu
TCON2	17h	0000 0000	0000 0000	uuuu uuuu
Unbanked				
PRODL ⁽⁵⁾	18h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PRODH ⁽⁵⁾	19h	xxxx xxxx	uuuu uuuu	uuuu uuuu

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

Note 1: One or more bits in INTSTA, PIR will be affected (to cause wake-up).

- 2: When the wake-up is due to an interrupt and the GLINTD bit is cleared, the PC is loaded with the interrupt vector.
- 3: See Table 4-3 for reset value of specific condition.
- 4: Only applies to the PIC17C42.
- 5: Does not apply to the PIC17C42.

5.0 INTERRUPTS

The PIC17C4X devices have 11 sources of interrupt:

- External interrupt from the RA0/INT pin
- Change on RB7:RB0 pins
- TMR0 Overflow
- TMR1 Overflow
- TMR2 Overflow
- TMR3 Overflow
- USART Transmit buffer empty
- USART Receive buffer full
- Capture1
- Capture2
- T0CKI edge occurred

There are four registers used in the control and status of interrupts. These are:

- CPUSTA
- INTSTA
- PIE
- PIR

The CPUSTA register contains the GLINTD bit. This is the Global Interrupt Disable bit. When this bit is set, all interrupts are disabled. This bit is part of the controller core functionality and is described in the Memory Organization section.

When an interrupt is responded to, the GLINTD bit is automatically set to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with the interrupt vector address. There are four interrupt vectors. Each vector address is for a specific interrupt source (except the peripheral interrupts which have the same vector address). These sources are:

- External interrupt from the RA0/INT pin
- TMR0 Overflow
- T0CKI edge occurred
- Any peripheral interrupt

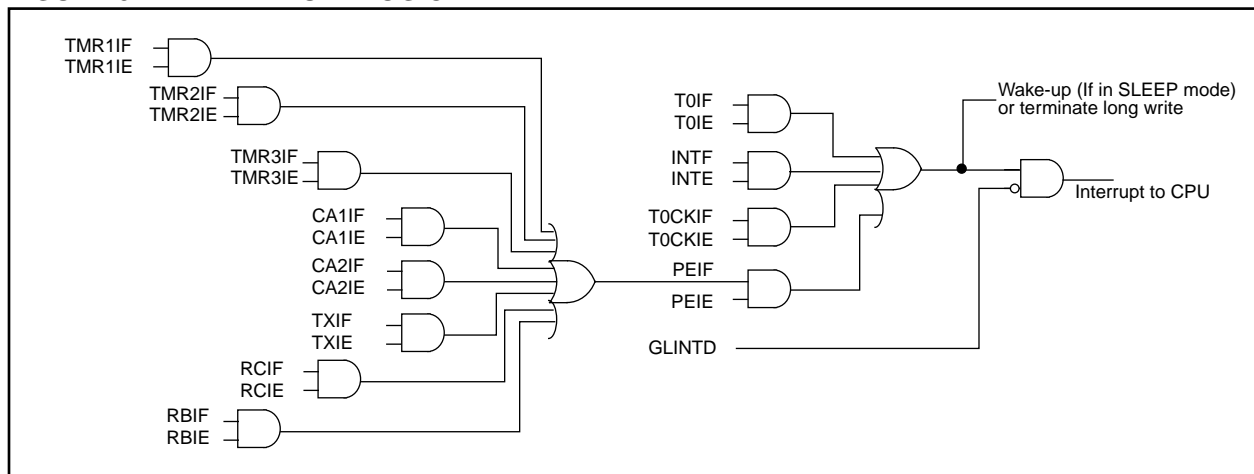
When program execution vectors to one of these interrupt vector addresses (except for the peripheral interrupt address), the interrupt flag bit is automatically cleared. Vectoring to the peripheral interrupt vector address does not automatically clear the source of the interrupt. In the peripheral interrupt service routine, the source(s) of the interrupt can be determined by testing the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid infinite interrupt requests.

All of the individual interrupt flag bits will be set regardless of the status of their corresponding mask bit or the GLINTD bit.

For external interrupt events, there will be an interrupt latency. For two cycle instructions, the latency could be one instruction cycle longer.

The “return from interrupt” instruction, `RETFIE`, can be used to mark the end of the interrupt service routine. When this instruction is executed, the stack is “POPed”, and the GLINTD bit is cleared (to re-enable interrupts).

FIGURE 5-1: INTERRUPT LOGIC



6.8 Bank Select Register (BSR)

The BSR is used to switch between banks in the data memory area (Figure 6-13). In the PIC17C42, PIC17CR42, and PIC17C42A only the lower nibble is implemented. While in the PIC17C43, PIC17CR43, and PIC17C44 devices, the entire byte is implemented. The lower nibble is used to select the peripheral register bank. The upper nibble is used to select the general purpose memory bank.

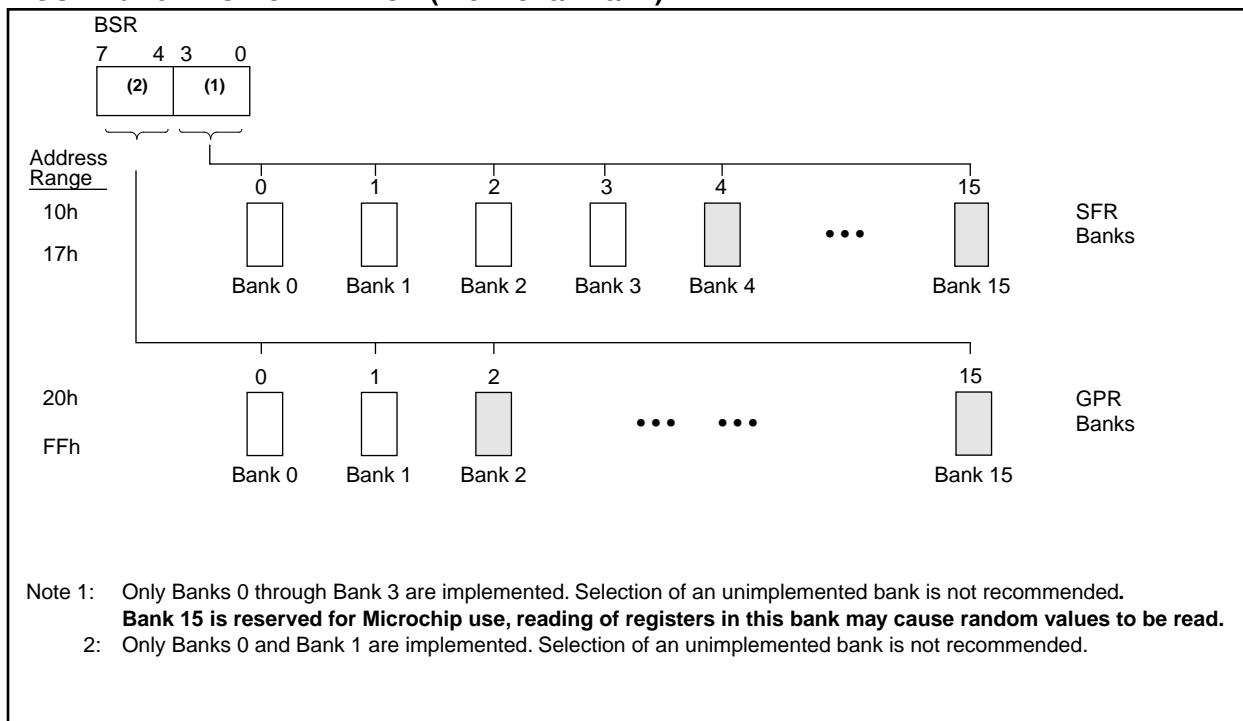
All the Special Function Registers (SFRs) are mapped into the data memory space. In order to accommodate the large number of registers, a banking scheme has been used. A segment of the SFRs, from address 10h to address 17h, is banked. The lower nibble of the bank select register (BSR) selects the currently active "peripheral bank." Effort has been made to group the peripheral registers of related functionality in one bank. However, it will still be necessary to switch from bank to bank in order to address all peripherals related to a single task. To assist this, a `MOVLB` bank instruction is in the instruction set.

For the PIC17C43, PIC17CR43, and PIC17C44 devices, the need for a large general purpose memory space dictated a general purpose RAM banking scheme. The upper nibble of the BSR selects the currently active general purpose RAM bank. To assist this, a `MOVLB` bank instruction has been provided in the instruction set.

If the currently selected bank is not implemented (such as Bank 13), any read will read all '0's. Any write is completed to the bit bucket and the ALU status bits will be set/cleared as appropriate.

Note: Registers in Bank 15 in the Special Function Register area, are reserved for Microchip use. Reading of registers in this bank may cause random values to be read.

FIGURE 6-13: BSR OPERATION (PIC17C43/R43/44)



Example 9-1 shows the instruction sequence to initialize PORTB. The Bank Select Register (BSR) must be selected to Bank 0 for the port to be initialized.

EXAMPLE 9-1: INITIALIZING PORTB

```
MOVLB 0      ; Select Bank 0
CLRF  PORTB  ; Initialize PORTB by clearing
              ; output data latches
MOVLW 0xCF   ; Value used to initialize
              ; data direction
MOVWF DDRB   ; Set RB<3:0> as inputs
              ; RB<5:4> as outputs
              ; RB<7:6> as inputs
```

TABLE 9-3: PORTB FUNCTIONS

Name	Bit	Buffer Type	Function
RB0/CAP1	bit0	ST	Input/Output or the RB0/CAP1 input pin. Software programmable weak pull-up and interrupt on change features.
RB1/CAP2	bit1	ST	Input/Output or the RB1/CAP2 input pin. Software programmable weak pull-up and interrupt on change features.
RB2/PWM1	bit2	ST	Input/Output or the RB2/PWM1 output pin. Software programmable weak pull-up and interrupt on change features.
RB3/PWM2	bit3	ST	Input/Output or the RB3/PWM2 output pin. Software programmable weak pull-up and interrupt on change features.
RB4/TCLK12	bit4	ST	Input/Output or the external clock input to Timer1 and Timer2. Software programmable weak pull-up and interrupt on change features.
RB5/TCLK3	bit5	ST	Input/Output or the external clock input to Timer3. Software programmable weak pull-up and interrupt on change features.
RB6	bit6	ST	Input/Output pin. Software programmable weak pull-up and interrupt on change features.
RB7	bit7	ST	Input/Output pin. Software programmable weak pull-up and interrupt on change features.

Legend: ST = Schmitt Trigger input.

TABLE 9-4: REGISTERS/BITS ASSOCIATED WITH PORTB

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)
12h, Bank 0	PORTB	PORTB data latch								xxxx xxxx	uuuu uuuu
11h, Bank 0	DDRB	Data direction register for PORTB								1111 1111	1111 1111
10h, Bank 0	PORTA	RBPU	—	RA5	RA4	RA3	RA2	RA1/T0CKI	RA0/INT	0-xx xxxx	0-uu uuuu
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	T0	PD	—	—	--11 11--	--11 qq--
07h, Unbanked	INTSTA	PEIF	T0CKIF	T0IF	INTF	PEIE	T0CKIE	T0IE	INTE	0000 0000	0000 0000
16h, Bank 1	PIR	RBIF	TMR3IF	TMR2IF	TMR1IF	CA2IF	CA1IF	TXIF	RCIF	0000 0010	0000 0010
17h, Bank 1	PIE	RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TXIE	RCIE	0000 0000	0000 0000
16h, Bank 3	TCON1	CA2ED1	CA2ED0	CA1ED1	CA1ED0	T16	TMR3CS	TMR2CS	TMR1CS	0000 0000	0000 0000
17h, Bank 3	TCON2	CA2OVF	CA1OVF	PWM2ON	PWM1ON	CA1/PR3	TMR3ON	TMR2ON	TMR1ON	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0', q = Value depends on condition.

Shaded cells are not used by PORTB.

Note 1: Other (non power-up) resets include: external reset through $\overline{\text{MCLR}}$ and the Watchdog Timer Reset.

FIGURE 13-2: RCSTA REGISTER (ADDRESS: 13h, BANK 0)

R/W - 0	R/W - 0	R/W - 0	R/W - 0	U - 0	R - 0	R - 0	R - x
SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D
bit 7							bit 0
<p>bit 7: SPEN: Serial Port Enable bit 1 = Configures RA5/RX/DT and RA4/TX/CK pins as serial port pins 0 = Serial port disabled</p> <p>bit 6: RX9: 9-bit Receive Enable bit 1 = Selects 9-bit reception 0 = Selects 8-bit reception</p> <p>bit 5: SREN: Single Receive Enable bit This bit enables the reception of a single byte. After receiving the byte, this bit is automatically cleared. <u>Synchronous mode</u>: 1 = Enable reception 0 = Disable reception Note: This bit is ignored in synchronous slave reception. <u>Asynchronous mode</u>: Don't care</p> <p>bit 4: CREN: Continuous Receive Enable bit This bit enables the continuous reception of serial data. <u>Asynchronous mode</u>: 1 = Enable reception 0 = Disables reception <u>Synchronous mode</u>: 1 = Enables continuous reception until CREN is cleared (CREN overrides SREN) 0 = Disables continuous reception</p> <p>bit 3: Unimplemented: Read as '0'</p> <p>bit 2: FERR: Framing Error bit 1 = Framing error (Updated by reading RCREG) 0 = No framing error</p> <p>bit 1: OERR: Overrun Error bit 1 = Overrun (Cleared by clearing CREN) 0 = No overrun error</p> <p>bit 0: RX9D: 9th bit of receive data (can be the software calculated parity bit)</p>							

R = Readable bit
W = Writable bit
-n = Value at POR reset
(x = unknown)

PIC17C4X

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

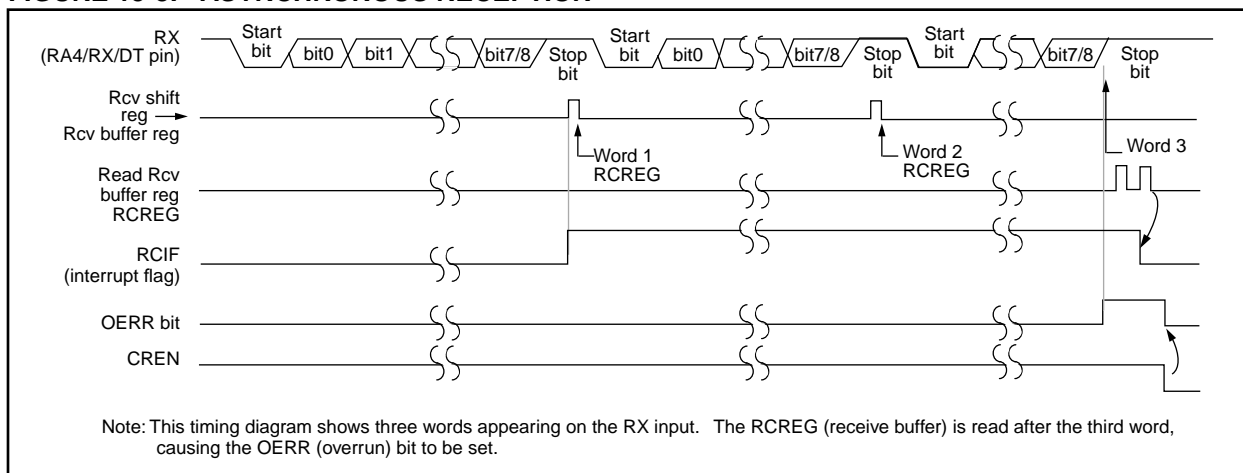


TABLE 13-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

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 MCLR and Watchdog Timer Reset.

TABLE 13-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER 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 synchronous master reception.

Note 1: Other (non power-up) resets include: external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

PIC17C4X

ANDWF		AND WREG with f							
Syntax:	[label] ANDWF f,d								
Operands:	$0 \leq f \leq 255$ $d \in [0,1]$								
Operation:	(WREG) .AND. (f) \rightarrow (dest)								
Status Affected:	Z								
Encoding:	<table border="1"><tr><td>0000</td><td>101d</td><td>ffff</td><td>ffff</td></tr></table>					0000	101d	ffff	ffff
0000	101d	ffff	ffff						
Description:	The contents of WREG are AND'ded 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:	[<i>label</i>] BCF f,b							
Operands:	$0 \leq f \leq 255$ $0 \leq b \leq 7$							
Operation:	$0 \rightarrow (f)$							
Status Affected:	None							
Encoding:	<table border="1"><tr><td>1000</td><td>1bbb</td><td>ffff</td><td>ffff</td></tr></table>				1000	1bbb	ffff	ffff
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

PIC17C4X

TABLRD Table Read

Example1: TABLRD 1, 1, REG ;

Before Instruction

REG = 0x53
TBLATH = 0xAA
TBLATL = 0x55
TBLPTR = 0xA356
MEMORY(TBLPTR) = 0x1234

After Instruction (table write completion)

REG = 0xAA
TBLATH = 0x12
TBLATL = 0x34
TBLPTR = 0xA357
MEMORY(TBLPTR) = 0x5678

Example2: TABLRD 0, 0, REG ;

Before Instruction

REG = 0x53
TBLATH = 0xAA
TBLATL = 0x55
TBLPTR = 0xA356
MEMORY(TBLPTR) = 0x1234

After Instruction (table write completion)

REG = 0x55
TBLATH = 0x12
TBLATL = 0x34
TBLPTR = 0xA356
MEMORY(TBLPTR) = 0x1234

TABLWT Table Write

Syntax: [label] TABLWT t,i,f

Operands: $0 \leq f \leq 255$
 $i \in [0,1]$
 $t \in [0,1]$

Operation: If $t = 0$,
 $f \rightarrow \text{TBLATL}$;
If $t = 1$,
 $f \rightarrow \text{TBLATH}$;
 $\text{TBLAT} \rightarrow \text{Prog Mem (TBLPTR)}$;
If $i = 1$,
 $\text{TBLPTR} + 1 \rightarrow \text{TBLPTR}$

Status Affected: None

Encoding:

1010	11ti	ffff	ffff
------	------	------	------

Description:

1. Load value in 'f' into 16-bit table latch (TBLAT)
If $t = 0$: load into low byte;
If $t = 1$: load into high byte
2. The contents of TBLAT is written to the program memory location pointed to by TBLPTR
If TBLPTR points to external program memory location, then the instruction takes two-cycle
If TBLPTR points to an internal EPROM location, then the instruction is terminated when an interrupt is received.

Note: The $\overline{\text{MCLR}}$ /VPP pin must be at the programming voltage for successful programming of internal memory.
If $\overline{\text{MCLR}}$ /VPP = VDD the programming sequence of internal memory will be executed, but will not be successful (although the internal memory location may be disturbed)

3. The TBLPTR can be automatically incremented
If $i = 0$; TBLPTR is not incremented
If $i = 1$; TBLPTR is incremented

Words: 1

Cycles: 2 (many if write is to on-chip EPROM program memory)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Execute	Write register TBLATH or TBLATL

PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

TABLE 17-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

OSC	PIC17C42-16	PIC17C42-25
RC	VDD: 4.5V to 5.5V IDD: 6 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 6 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 4 MHz max.
XT	VDD: 4.5V to 5.5V IDD: 24 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 16 MHz max.	VDD: 4.5V to 5.5V IDD: 38 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 25 MHz max.
EC	VDD: 4.5V to 5.5V IDD: 24 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 16 MHz max.	VDD: 4.5V to 5.5V IDD: 38 mA max. IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 25 MHz max.
LF	VDD: 4.5V to 5.5V IDD: 150 μ A max. at 32 kHz (WDT enabled) IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 2 MHz max.	VDD: 4.5V to 5.5V IDD: 150 μ A max. at 32 kHz (WDT enabled) IPD: 5 μ A max. at 5.5V (WDT disabled) Freq: 2 MHz max.

PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

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

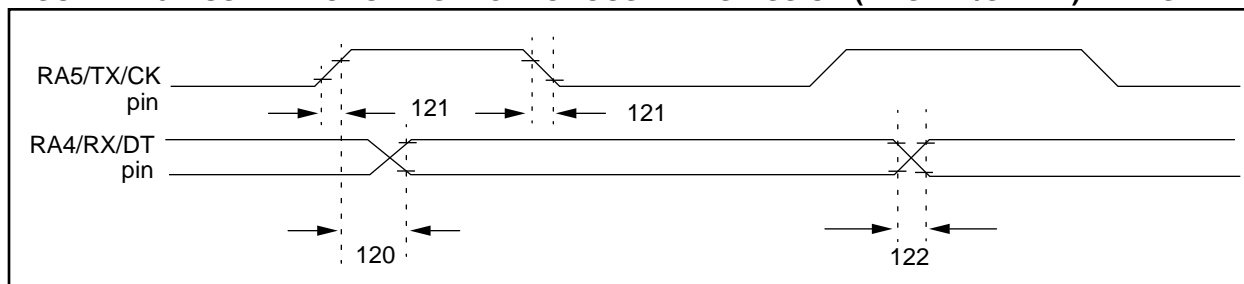


TABLE 17-9: SERIAL PORT SYNCHRONOUS TRANSMISSION REQUIREMENTS

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

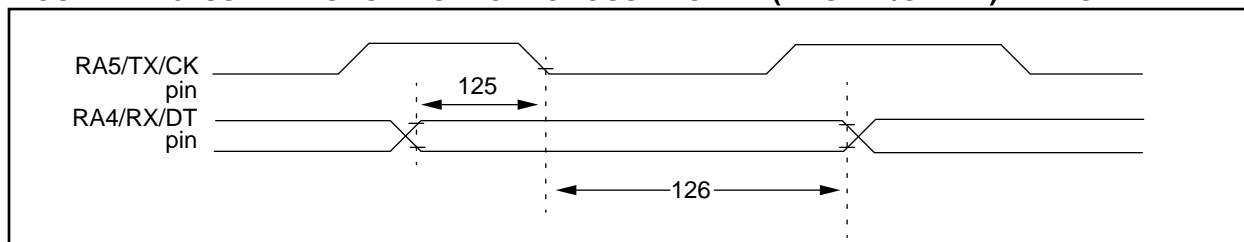


TABLE 17-10: SERIAL PORT SYNCHRONOUS RECEIVE REQUIREMENTS

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.

PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

FIGURE 18-13: WDT TIMER TIME-OUT PERIOD vs. VDD

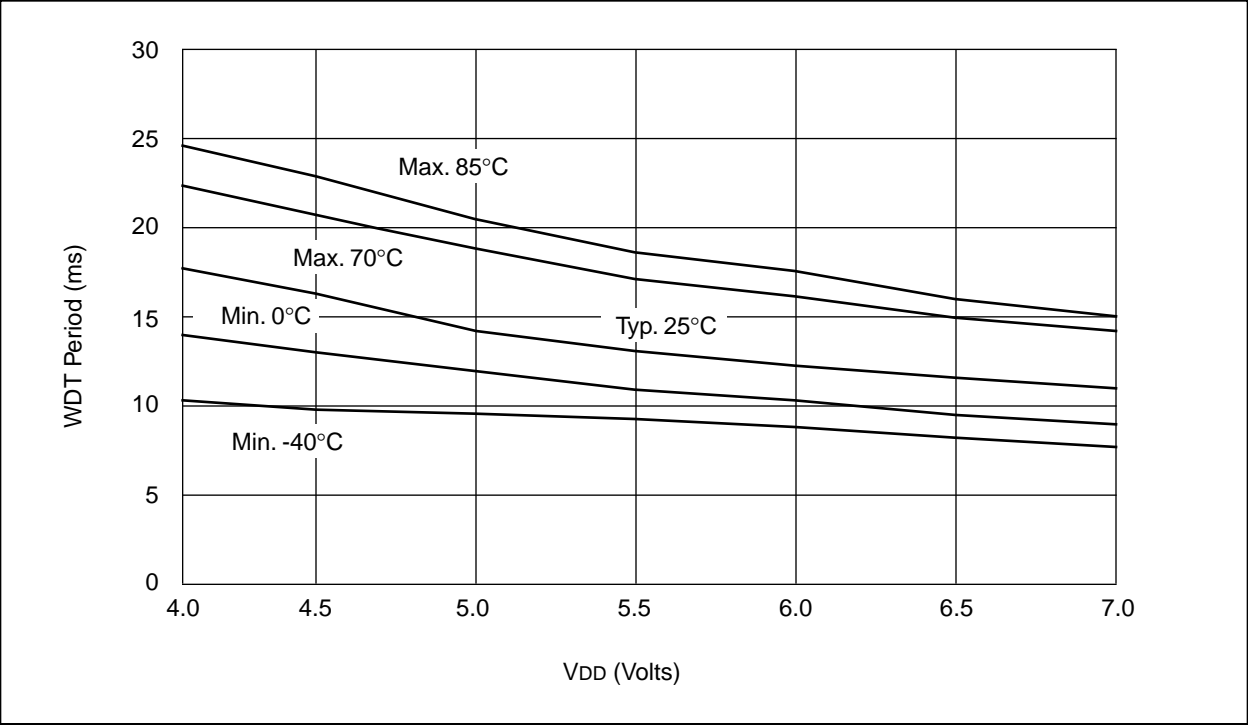
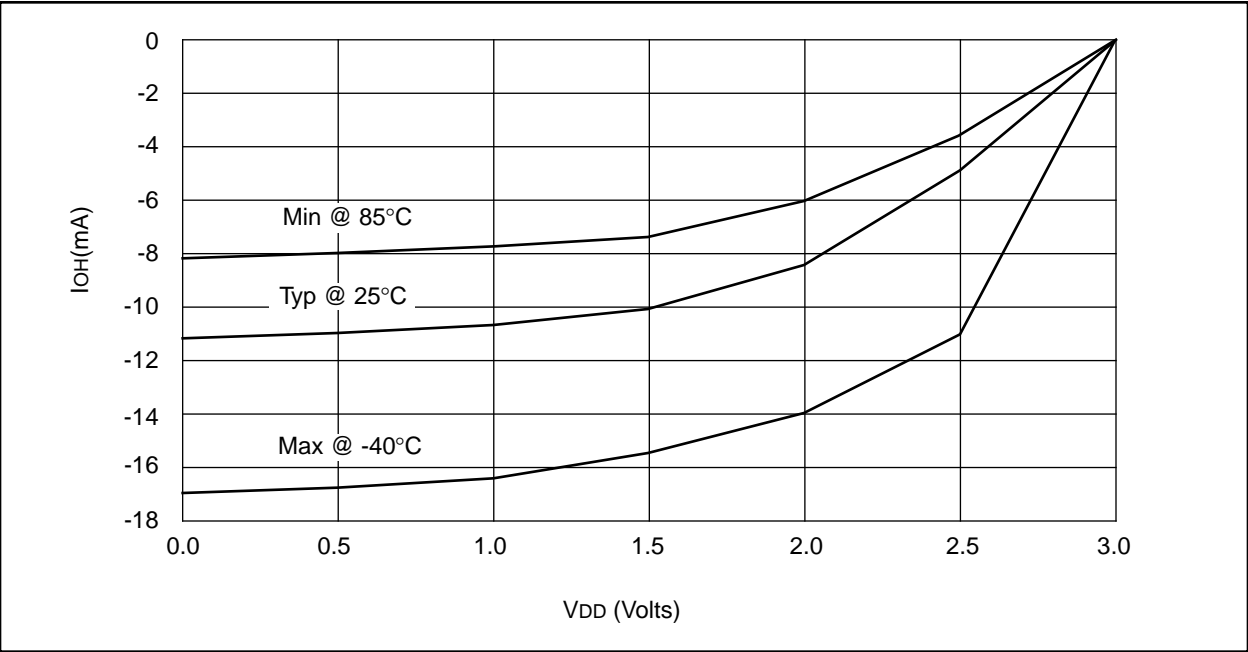


FIGURE 18-14: IOH vs. VOH, VDD = 3V



PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

FIGURE 19-7: CAPTURE TIMINGS

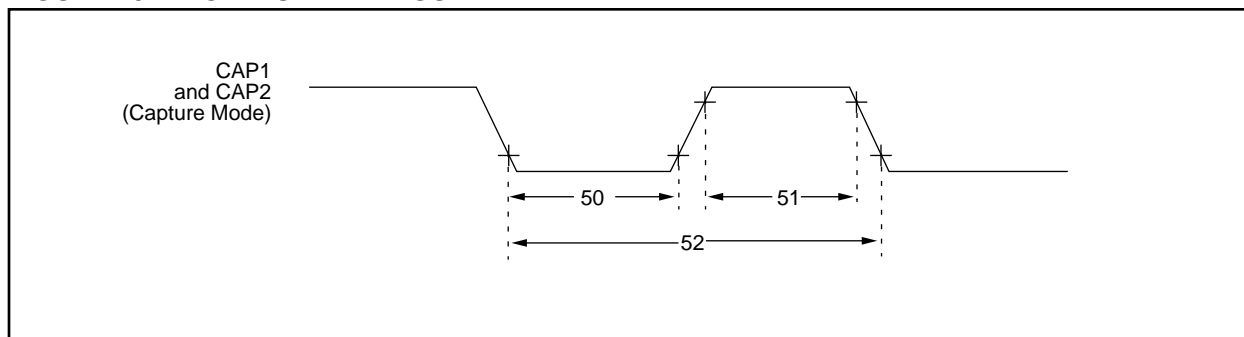


TABLE 19-7: CAPTURE REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
50	TccL	Capture1 and Capture2 input low time	10 *	—	—	ns	
51	TccH	Capture1 and Capture2 input high time	10 *	—	—	ns	
52	TccP	Capture1 and Capture2 input period	$\frac{2T_{CY}}{N}$ §	—	—	ns	N = prescale value (4 or 16)

* 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.

FIGURE 19-8: PWM TIMINGS

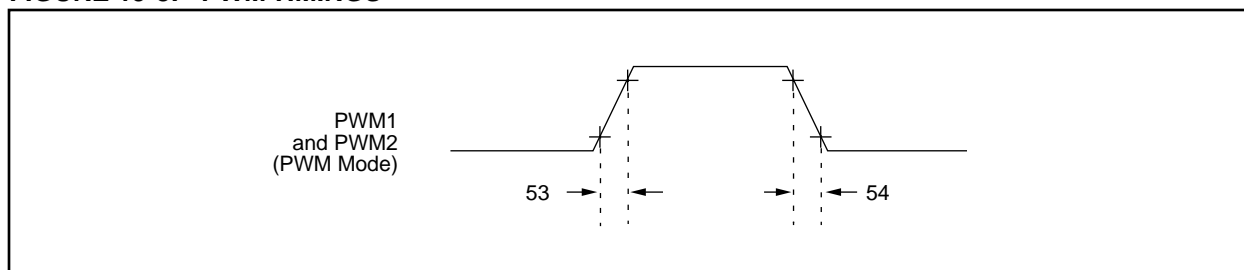


TABLE 19-8: PWM REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
53	TccR	PWM1 and PWM2 output rise time	—	10 *	35 *§	ns	
54	TccF	PWM1 and PWM2 output fall time	—	10 *	35 *§	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.

FIGURE 19-9: USART MODULE: SYNCHRONOUS TRANSMISSION (MASTER/S�AVE) TIMING

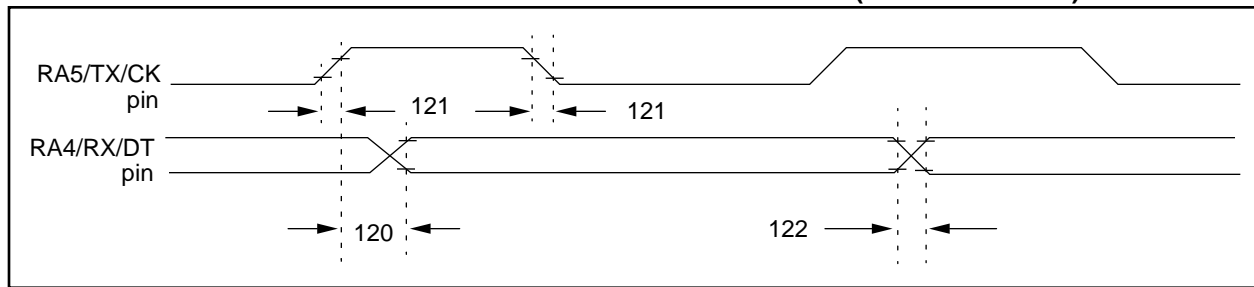


TABLE 19-9: SYNCHRONOUS TRANSMISSION REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE) Clock high to data out valid	PIC17CR42/42A/43/R43/44	—	—	50	ns
			PIC17LCR42/42A/43/R43/44	—	—	75	ns
121	TckRF	Clock out rise time and fall time (Master Mode)	PIC17CR42/42A/43/R43/44	—	—	25	ns
			PIC17LCR42/42A/43/R43/44	—	—	40	ns
122	TdtRF	Data out rise time and fall time	PIC17CR42/42A/43/R43/44	—	—	25	ns
			PIC17LCR42/42A/43/R43/44	—	—	40	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 19-10: USART MODULE: SYNCHRONOUS RECEIVE (MASTER/S�AVE) TIMING

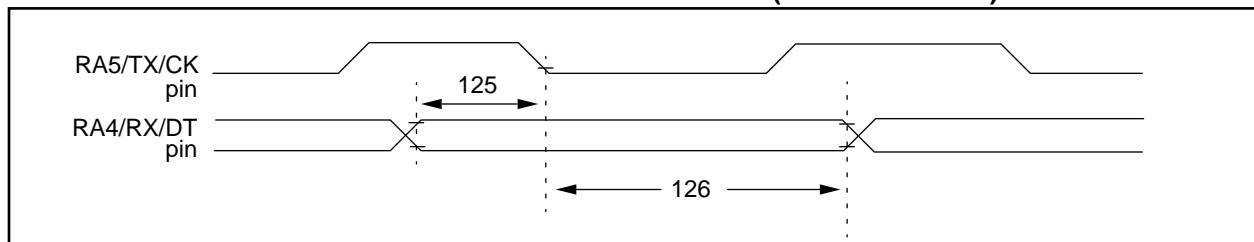


TABLE 19-10: SYNCHRONOUS RECEIVE REQUIREMENTS

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	TckL2dtI	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.

FIGURE 19-12: MEMORY INTERFACE READ TIMING (NOT SUPPORTED IN PIC17LC4X DEVICES)

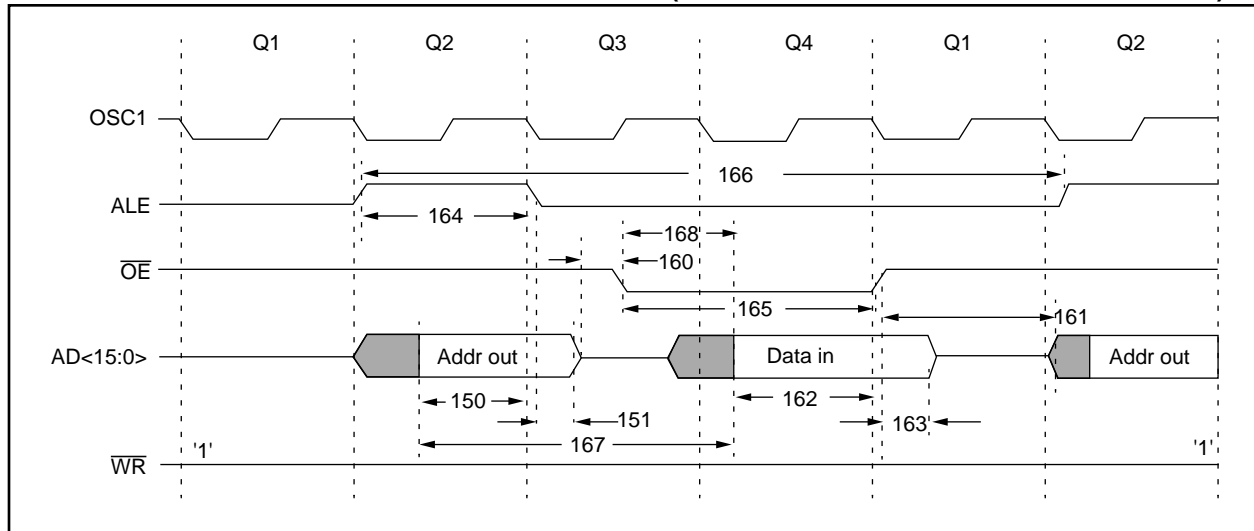


TABLE 19-12: MEMORY INTERFACE READ REQUIREMENTS (NOT SUPPORTED IN PIC17LC4X DEVICES)

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
150	TadV2alL	AD15:AD0 (address) valid to ALE↓ (address setup time)	0.25Tcy - 10	—	—	ns	
151	TalL2adI	ALE↓ to address out invalid (address hold time)	5*	—	—	ns	
160	TadZ2oeL	AD15:AD0 hi-impedance to \overline{OE} ↓	0*	—	—	ns	
161	ToeH2adD	\overline{OE} ↑ to AD15:AD0 driven	0.25Tcy - 15	—	—	ns	
162	TadV2oeH	Data in valid before \overline{OE} ↑ (data setup time)	35	—	—	ns	
163	ToeH2adI	\overline{OE} ↑ to data in invalid (data hold time)	0	—	—	ns	
164	TalH	ALE pulse width	—	0.25Tcy §	—	ns	
165	ToeL	\overline{OE} pulse width	0.5Tcy - 35 §	—	—	ns	
166	TalH2alH	ALE↑ to ALE↑ (cycle time)	—	Tcy §	—	ns	
167	Tacc	Address access time	—	—	0.75Tcy - 30	ns	
168	Toe	Output enable access time (\overline{OE} low to Data Valid)	—	—	0.5Tcy - 45	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.

PIC17C4X

Delay From External Clock Edge	68
Development Support	143
Development Tools	143
Device Drawings	
44-Lead Plastic Surface Mount (MQFP	
10x10 mm Body 1.6/0.15 mm Lead Form)	209
DIGIT BORROW	9
Digit Carry (DC)	9
Duty Cycle	75

E

Electrical Characteristics

PIC17C42	
Absolute Maximum Ratings	147
Capture Timing	159
CLKOUT and I/O Timing	156
DC Characteristics	149
External Clock Timing	155
Memory Interface Read Timing	162
Memory Interface Write Timing	161
PWM Timing	159
RESET, Watchdog Timer, Oscillator Start-up	
Timer and Power-up Timer	157
Timer0 Clock Timings	158
Timer1, Timer2 and Timer3 Clock Timing	158
USART Module, Synchronous Receive	160
USART Module, Synchronous Transmission	160
PIC17C43/44	
Absolute Maximum Ratings	175
Capture Timing	188
CLKOUT and I/O Timing	185
DC Characteristics	177
External Clock Timing	184
Memory Interface Read Timing	191
Memory Interface Write Timing	190
Parameter Measurement Information	183
RESET, Watchdog Timer, Oscillator Start-up	
Timer and Power-up Timer Timing	186
Timer0 Clock Timing	187
Timer1, Timer2 and Timer3 Clock Timing	187
Timing Parameter Symbolology	182
USART Module Synchronous Receive	
Timing	189
USART Module Synchronous Transmission	
Timing	189
EPROM Memory Access Time Order Suffix	31
Extended Microcontroller	29
Extended Microcontroller Mode	31
External Memory Interface	31
External Program Memory Waveforms	31

F

Family of Devices	6
PIC14000	213
PIC16C5X	214
PIC16CXXX	215
PIC16C6X	216
PIC16C7X	217
PIC16C8X	218
PIC16C9XX	219
PIC17CXX	220
FERR	84, 91
FOSC0	99

FOSC1	99
FS0	36
FS1	36
FS2	36
FS3	36
FSR0	34, 40
FSR1	34, 40
Fuzzy Logic Dev. System (<i>fuzzyTECH</i> ®-MP)	143, 145

G

General Format for Instructions	108
General Purpose RAM	29
General Purpose RAM Bank	42
General Purpose Register (GPR)	32
GLINTD	25, 37, 78, 105
GOTO	122
GPR (General Purpose Register)	32
Graphs	
IOH vs. VOH, VDD = 3V	170, 200
IOH vs. VOH, VDD = 5V	171, 201
IOL vs. VOL, VDD = 3V	171, 201
IOL vs. VOL, VDD = 5V	172, 202
Maximum IDD vs. Frequency	
(External Clock 125°C to -40°C)	167, 197
Maximum IPD vs. VDD Watchdog Disabled	168, 198
Maximum IPD vs. VDD Watchdog Enabled	169, 199
RC Oscillator Frequency vs.	
VDD (Cext = 100 pF)	164, 194
RC Oscillator Frequency vs.	
VDD (Cext = 22 pF)	164, 194
RC Oscillator Frequency vs.	
VDD (Cext = 300 pF)	165, 195
Transconductance of LF Oscillator vs. VDD	166, 196
Transconductance of XT Oscillator vs. VDD	166, 196
Typical IDD vs. Frequency	
(External Clock 25°C)	167, 197
Typical IPD vs. VDD Watchdog Disabled 25°C	168, 198
Typical IPD vs. VDD Watchdog Enabled 25°C	169, 199
Typical RC Oscillator vs. Temperature	163, 193
VTH (Input Threshold Voltage) of I/O Pins vs.	
VDD	172, 202
VTH (Input Threshold Voltage) of OSC1 Input	
(In XT, HS, and LP Modes) vs. VDD	173, 203
VTH, VIL of MCLR, T0CKI and OSC1	
(In RC Mode) vs. VDD	173, 203
WDT Timer Time-Out Period vs. VDD	170, 200

H

Hardware Multiplier	49
---------------------------	----

I

I/O Ports	
Bi-directional	64
I/O Ports	53
Programming Considerations	64
Read-Modify-Write Instructions	64
Successive Operations	64
INCF	123
INCFSNZ	124
INCFSZ	123
INDF0	34, 40
INDF1	34, 40

Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable”.
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

Trademarks


The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

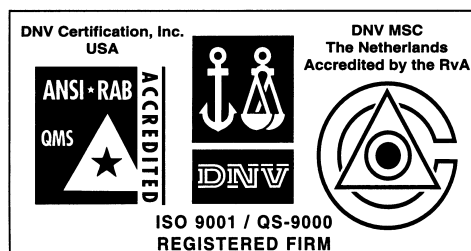
dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.



Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.