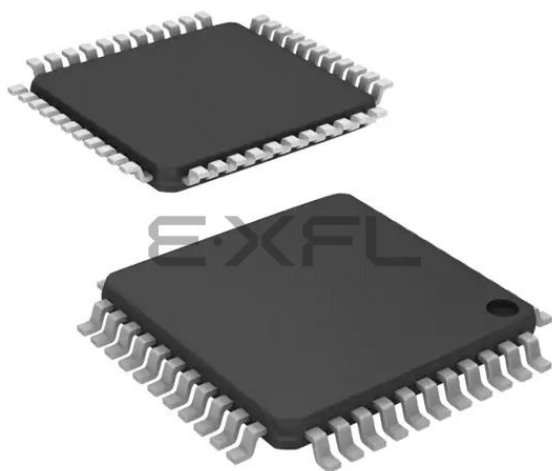




Welcome to [E-XFL.COM](https://www.e-xfl.com)

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	16MHz
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 ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c43-16i-pt

PIC17C4X

TABLE 3-1: PINOUT DESCRIPTIONS

Name	DIP No.	PLCC No.	QFP No.	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	19	21	37	I	ST	Oscillator input in crystal/resonator or RC oscillator mode. External clock input in external clock mode.
OSC2/CLKOUT	20	22	38	O	—	Oscillator output. Connects to crystal or resonator in crystal oscillator mode. In RC oscillator or external clock modes OSC2 pin outputs CLKOUT which has one fourth the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP	32	35	7	I/P	ST	Master clear (reset) input/Programming Voltage (VPP) input. This is the active low reset input to the chip.
RA0/INT	26	28	44	I	ST	PORTA is a bi-directional I/O Port except for RA0 and RA1 which are input only. RA0/INT can also be selected as an external interrupt input. Interrupt can be configured to be on positive or negative edge. RA1/T0CKI can also be selected as an external interrupt input, and the interrupt can be configured to be on positive or negative edge. RA1/T0CKI can also be selected to be the clock input to the Timer0 timer/counter. High voltage, high current, open drain input/output port pins. High voltage, high current, open drain input/output port pins. RA4/RX/DT can also be selected as the USART (SCI) Asynchronous Receive or USART (SCI) Synchronous Data. RA5/TX/CK can also be selected as the USART (SCI) Asynchronous Transmit or USART (SCI) Synchronous Clock.
RA1/T0CKI	25	27	43	I	ST	
RA2	24	26	42	I/O	ST	
RA3	23	25	41	I/O	ST	
RA4/RX/DT	22	24	40	I/O	ST	
RA5/TX/CK	21	23	39	I/O	ST	
RB0/CAP1	11	13	29	I/O	ST	PORTB is a bi-directional I/O Port with software configurable weak pull-ups. RB0/CAP1 can also be the CAP1 input pin. RB1/CAP2 can also be the CAP2 input pin. RB2/PWM1 can also be the PWM1 output pin. RB3/PWM2 can also be the PWM2 output pin. RB4/TCLK12 can also be the external clock input to Timer1 and Timer2. RB5/TCLK3 can also be the external clock input to Timer3.
RB1/CAP2	12	14	30	I/O	ST	
RB2/PWM1	13	15	31	I/O	ST	
RB3/PWM2	14	16	32	I/O	ST	
RB4/TCLK12	15	17	33	I/O	ST	
RB5/TCLK3	16	18	34	I/O	ST	
RB6	17	19	35	I/O	ST	
RB7	18	20	36	I/O	ST	
RC0/AD0	2	3	19	I/O	TTL	PORTC is a bi-directional I/O Port. This is also the lower half of the 16-bit wide system bus in microprocessor mode or extended microcontroller mode. In multiplexed system bus configuration, these pins are address output as well as data input or output.
RC1/AD1	3	4	20	I/O	TTL	
RC2/AD2	4	5	21	I/O	TTL	
RC3/AD3	5	6	22	I/O	TTL	
RC4/AD4	6	7	23	I/O	TTL	
RC5/AD5	7	8	24	I/O	TTL	
RC6/AD6	8	9	25	I/O	TTL	
RC7/AD7	9	10	26	I/O	TTL	

Legend: I = Input only; O = Output only; I/O = Input/Output; P = Power; — = Not Used; TTL = TTL input; ST = Schmitt Trigger input.

TABLE 4-4: INITIALIZATION CONDITIONS FOR SPECIAL FUNCTION REGISTERS

Register	Address	Power-on Reset	MCLR Reset WDT Reset	Wake-up from SLEEP through interrupt
Unbanked				
INDF0	00h	0000 0000	0000 0000	0000 0000
FSR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h	0000h	0000h	PC + 1 ⁽²⁾
PCLATH	03h	0000 0000	0000 0000	uuuu uuuu
ALUSTA	04h	1111 xxxx	1111 uuuu	1111 uuuu
T0STA	05h	0000 000-	0000 000-	0000 000-
CPUSTA ⁽³⁾	06h	--11 11--	--11 qq--	--uu qq--
INTSTA	07h	0000 0000	0000 0000	uuuu uuuu ⁽¹⁾
INDF1	08h	0000 0000	0000 0000	uuuu uuuu
FSR1	09h	xxxx xxxx	uuuu uuuu	uuuu uuuu
WREG	0Ah	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0L	0Bh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0H	0Ch	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRL ⁽⁴⁾	0Dh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRH ⁽⁴⁾	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRL ⁽⁵⁾	0Dh	0000 0000	0000 0000	uuuu uuuu
TBLPTRH ⁽⁵⁾	0Eh	0000 0000	0000 0000	uuuu uuuu
BSR	0Fh	0000 0000	0000 0000	uuuu uuuu
Bank 0				
PORTA	10h	0-xx xxxx	0-uu uuuu	uuuu uuuu
DDRB	11h	1111 1111	1111 1111	uuuu uuuu
PORTB	12h	xxxx xxxx	uuuu uuuu	uuuu uuuu
RCSTA	13h	0000 -00x	0000 -00u	uuuu -uuu
RCREG	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXSTA	15h	0000 --1x	0000 --1u	uuuu --uu
TXREG	16h	xxxx xxxx	uuuu uuuu	uuuu uuuu
SPBRG	17h	xxxx xxxx	uuuu uuuu	uuuu uuuu
Bank 1				
DDRC	10h	1111 1111	1111 1111	uuuu uuuu
PORTC	11h	xxxx xxxx	uuuu uuuu	uuuu uuuu
DDRD	12h	1111 1111	1111 1111	uuuu uuuu
PORTD	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
DDRE	14h	---- -111	---- -111	---- -uuu
PORTE	15h	---- -xxx	---- -uuu	---- -uuu
PIR	16h	0000 0010	0000 0010	uuuu uuuu ⁽¹⁾
PIE	17h	0000 0000	0000 0000	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.

6.3 Stack Operation

The PIC17C4X devices have a 16 x 16-bit wide hardware stack (Figure 6-1). The stack is not part of either the program or data memory space, and the stack pointer is neither readable nor writable. The PC is “PUSHed” onto the stack when a CALL instruction is executed or an interrupt is acknowledged. The stack is “POPped” in the event of a RETURN, RETLW, or a RETFIE instruction execution. PCLATH is not affected by a “PUSH” or a “POP” operation.

The stack operates as a circular buffer, with the stack pointer initialized to '0' after all resets. There is a stack available bit (STKAV) to allow software to ensure that the stack has not overflowed. The STKAV bit is set after a device reset. When the stack pointer equals Fh, STKAV is cleared. When the stack pointer rolls over from Fh to 0h, the STKAV bit will be held clear until a device reset.

- Note 1:** There is not a status bit for stack underflow. The STKAV bit can be used to detect the underflow which results in the stack pointer being at the top of stack.
- Note 2:** There are no instruction mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW, and RETFIE instructions, or the vectoring to an interrupt vector.
- Note 3:** After a reset, if a “POP” operation occurs before a “PUSH” operation, the STKAV bit will be cleared. This will appear as if the stack is full (underflow has occurred). If a “PUSH” operation occurs next (before another “POP”), the STKAV bit will be locked clear. Only a device reset will cause this bit to set.

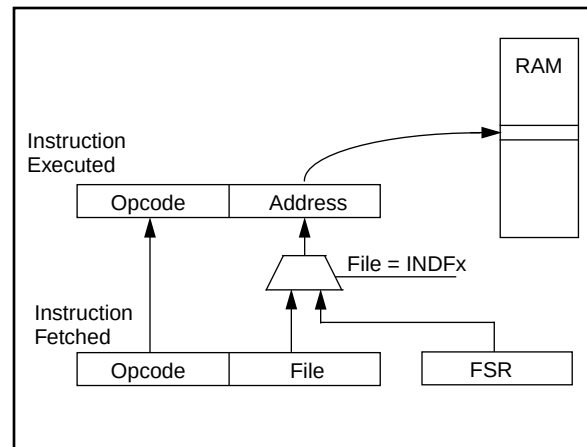
After the device is “PUSHed” sixteen times (without a “POP”), the seventeenth push overwrites the value from the first push. The eighteenth push overwrites the second push (and so on).

6.4 Indirect Addressing

Indirect addressing is a mode of addressing data memory where the data memory address in the instruction is not fixed. That is, the register that is to be read or written can be modified by the program. This can be useful for data tables in the data memory. Figure 6-10 shows the operation of indirect addressing. This shows the moving of the value to the data memory address specified by the value of the FSR register.

Example 6-1 shows the use of indirect addressing to clear RAM in a minimum number of instructions. A similar concept could be used to move a defined number of bytes (block) of data to the USART transmit register (TXREG). The starting address of the block of data to be transmitted could easily be modified by the program.

FIGURE 6-10: INDIRECT ADDRESSING



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 7-3: TLRD INSTRUCTION OPERATION

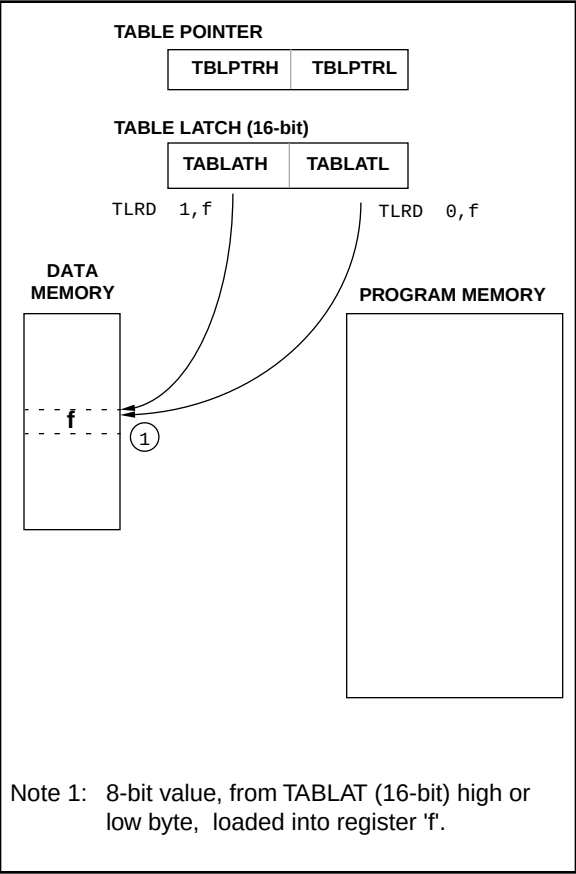


FIGURE 7-4: TABLRD INSTRUCTION OPERATION

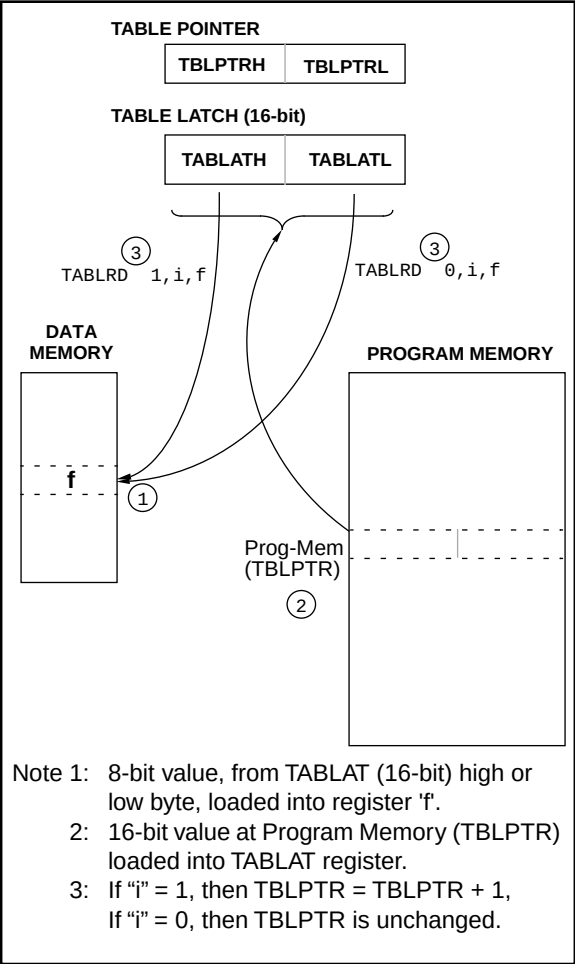


TABLE 9-5: PORTC FUNCTIONS

Name	Bit	Buffer Type	Function
RC0/AD0	bit0	TTL	Input/Output or system bus address/data pin.
RC1/AD1	bit1	TTL	Input/Output or system bus address/data pin.
RC2/AD2	bit2	TTL	Input/Output or system bus address/data pin.
RC3/AD3	bit3	TTL	Input/Output or system bus address/data pin.
RC4/AD4	bit4	TTL	Input/Output or system bus address/data pin.
RC5/AD5	bit5	TTL	Input/Output or system bus address/data pin.
RC6/AD6	bit6	TTL	Input/Output or system bus address/data pin.
RC7/AD7	bit7	TTL	Input/Output or system bus address/data pin.

Legend: TTL = TTL input.

TABLE 9-6: REGISTERS/BITS ASSOCIATED WITH PORTC

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)
11h, Bank 1	PORTC	RC7/ AD7	RC6/ AD6	RC5/ AD5	RC4/ AD4	RC3/ AD3	RC2/ AD2	RC1/ AD1	RC0/ AD0	xxxx xxxx	uuuu uuuu
10h, Bank 1	DDRC	Data direction register for PORTC								1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

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

10.0 OVERVIEW OF TIMER RESOURCES

The PIC17C4X has four timer modules. Each module can generate an interrupt to indicate that an event has occurred. These timers are called:

- Timer0 - 16-bit timer with programmable 8-bit prescaler
- Timer1 - 8-bit timer
- Timer2 - 8-bit timer
- Timer3 - 16-bit timer

For enhanced time-base functionality, two input Captures and two Pulse Width Modulation (PWM) outputs are possible. The PWMs use the TMR1 and TMR2 resources and the input Captures use the TMR3 resource.

10.1 Timer0 Overview

The Timer0 module is a simple 16-bit overflow counter. The clock source can be either the internal system clock ($F_{osc}/4$) or an external clock.

The Timer0 module also has a programmable prescaler option. The PS3:PS0 bits (T0STA<4:1>) determine the prescaler value. TMR0 can increment at the following rates: 1:1, 1:2, 1:4, 1:8, 1:16, 1:32, 1:64, 1:128, 1:256.

When Timer0's clock source is an external clock, the Timer0 module can be selected to increment on either the rising or falling edge.

Synchronization of the external clock occurs after the prescaler. When the prescaler is used, the external clock frequency may be higher than the device's frequency. The maximum frequency is 50 MHz, given the high and low time requirements of the clock.

10.2 Timer1 Overview

The Timer1 module is an 8-bit timer/counter with an 8-bit period register (PR1). When the TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB4/TCLK12 pin, which can also be selected to be the clock for the Timer2 module.

TMR1 can be concatenated to TMR2 to form a 16-bit timer. The TMR1 register is the LSB and TMR2 is the MSB. When in the 16-bit timer mode, there is a corresponding 16-bit period register (PR2:PR1). When the TMR2:TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled.

10.3 Timer2 Overview

The TMR2 module is an 8-bit timer/counter with an 8-bit period register (PR2). When the TMR2 value rolls over from the period match value to 0h, the TMR2IF flag is set, and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB4/TCLK12 pin, which can also be selected to be the clock for the TMR1 module.

TMR1 can be concatenated to TMR2 to form a 16-bit timer. The TMR2 register is the MSB and TMR1 is the LSB. When in the 16-bit timer mode, there is a corresponding 16-bit period register (PR2:PR1). When the TMR2:TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled.

10.4 Timer3 Overview

The Timer3 module is a 16-bit timer/counter with a 16-bit period register. When the TMR3H:TMR3L value rolls over to 0h, the TMR3IF bit is set and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB5/TCLK3 pin.

When operating in the dual capture mode, the period registers become the second 16-bit capture register.

10.5 Role of the Timer/Counters

The timer modules are general purpose, but have dedicated resources associated with them. Timer1 and Timer2 are the time-bases for the two Pulse Width Modulation (PWM) outputs, while Timer3 is the time-base for the two input captures.

12.1.3 USING PULSE WIDTH MODULATION (PWM) OUTPUTS WITH TMR1 AND TMR2

Two high speed pulse width modulation (PWM) outputs are provided. The PWM1 output uses Timer1 as its time-base, while PWM2 may be software configured to use either Timer1 or Timer2 as the time-base. The PWM outputs are on the RB2/PWM1 and RB3/PWM2 pins.

Each PWM output has a maximum resolution of 10-bits. At 10-bit resolution, the PWM output frequency is 24.4 kHz (@ 25 MHz clock) and at 8-bit resolution the PWM output frequency is 97.7 kHz. The duty cycle of the output can vary from 0% to 100%.

Figure 12-5 shows a simplified block diagram of the PWM module. The duty cycle register is double buffered for glitch free operation. Figure 12-6 shows how a glitch could occur if the duty cycle registers were not double buffered.

The user needs to set the PWM1ON bit (TCON2<4>) to enable the PWM1 output. When the PWM1ON bit is set, the RB2/PWM1 pin is configured as PWM1 output and forced as an output irrespective of the data direction bit (DDRB<2>). When the PWM1ON bit is clear, the pin behaves as a port pin and its direction is controlled by its data direction bit (DDRB<2>). Similarly, the PWM2ON (TCON2<5>) bit controls the configuration of the RB3/PWM2 pin.

FIGURE 12-5: SIMPLIFIED PWM BLOCK DIAGRAM

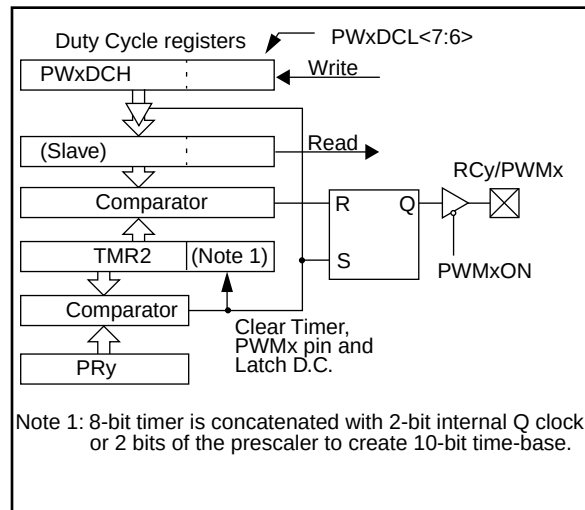


FIGURE 12-6: PWM OUTPUT

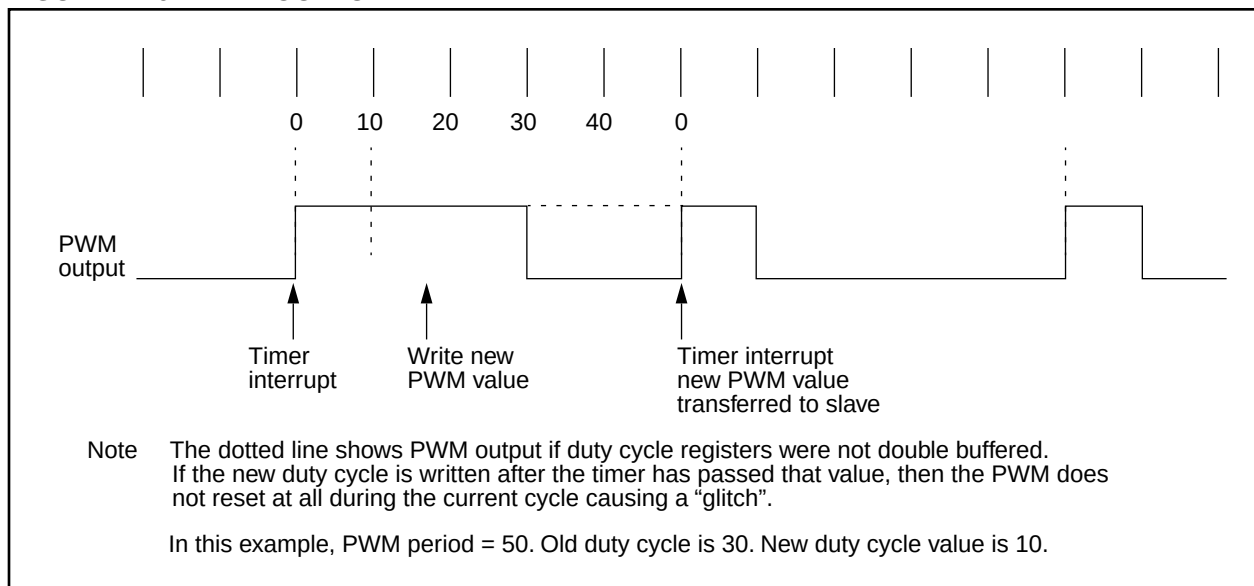


TABLE 13-3: BAUD RATES FOR SYNCHRONOUS MODE

BAUD RATE (K)	FOSC = 33 MHz			FOSC = 25 MHz			FOSC = 20 MHz			FOSC = 16 MHz		
	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)
0.3	NA	—	—	NA	—	—	NA	—	—	NA	—	—
1.2	NA	—	—	NA	—	—	NA	—	—	NA	—	—
2.4	NA	—	—	NA	—	—	NA	—	—	NA	—	—
9.6	NA	—	—	NA	—	—	NA	—	—	NA	—	—
19.2	NA	—	—	NA	—	—	19.53	+1.73	255	19.23	+0.16	207
76.8	77.10	+0.39	106	77.16	+0.47	80	76.92	+0.16	64	76.92	+0.16	51
96	95.93	-0.07	85	96.15	+0.16	64	96.15	+0.16	51	95.24	-0.79	41
300	294.64	-1.79	27	297.62	-0.79	20	294.1	-1.96	16	307.69	+2.56	12
500	485.29	-2.94	16	480.77	-3.85	12	500	0	9	500	0	7
HIGH	8250	—	0	6250	—	0	5000	—	0	4000	—	0
LOW	32.22	—	255	24.41	—	255	19.53	—	255	15.625	—	255

BAUD RATE (K)	FOSC = 10 MHz			FOSC = 7.159 MHz			FOSC = 5.068 MHz		
	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)
0.3	NA	—	—	NA	—	—	NA	—	—
1.2	NA	—	—	NA	—	—	NA	—	—
2.4	NA	—	—	NA	—	—	NA	—	—
9.6	9.766	+1.73	255	9.622	+0.23	185	9.6	0	131
19.2	19.23	+0.16	129	19.24	+0.23	92	19.2	0	65
76.8	75.76	-1.36	32	77.82	+1.32	22	79.2	+3.13	15
96	96.15	+0.16	25	94.20	-1.88	18	97.48	+1.54	12
300	312.5	+4.17	7	298.3	-0.57	5	316.8	+5.60	3
500	500	0	4	NA	—	—	NA	—	—
HIGH	2500	—	0	1789.8	—	0	1267	—	0
LOW	9.766	—	255	6.991	—	255	4.950	—	255

BAUD RATE (K)	Fosc = 3.579 MHz			FOSC = 1 MHz			FOSC = 32.768 kHz		
	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)
0.3	NA	—	—	NA	—	—	0.303	+1.14	26
1.2	NA	—	—	1.202	+0.16	207	1.170	-2.48	6
2.4	NA	—	—	2.404	+0.16	103	NA	—	—
9.6	9.622	+0.23	92	9.615	+0.16	25	NA	—	—
19.2	19.04	-0.83	46	19.24	+0.16	12	NA	—	—
76.8	74.57	-2.90	11	83.34	+8.51	2	NA	—	—
96	99.43	-3.57	8	NA	—	—	NA	—	—
300	298.3	-0.57	2	NA	—	—	NA	—	—
500	NA	—	—	NA	—	—	NA	—	—
HIGH	894.9	—	0	250	—	0	8.192	—	0
LOW	3.496	—	255	0.976	—	255	0.032	—	255

FIGURE 14-8: WATCHDOG TIMER BLOCK DIAGRAM

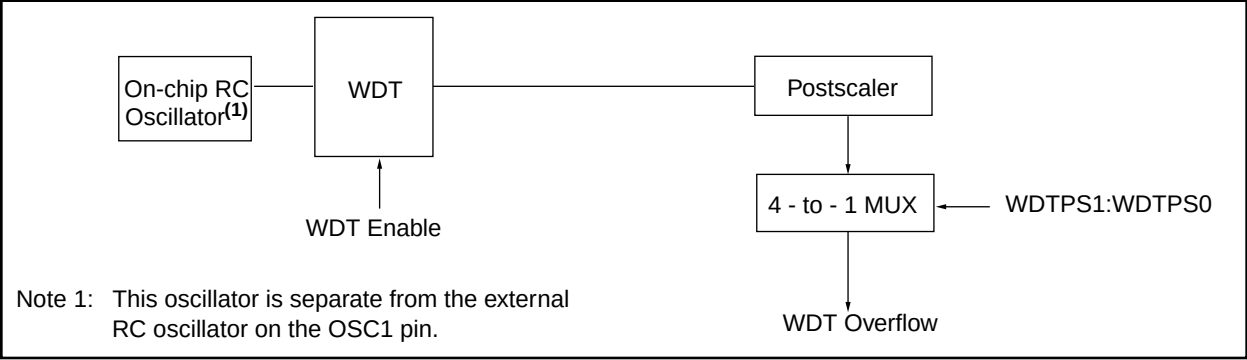


TABLE 14-4: REGISTERS/BITS ASSOCIATED WITH THE WATCHDOG TIMER

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)
—	Config	—	PM1	—	PM0	WDTPS1	WDTPS0	FOSC1	FOSC0	(Note 2)	(Note 2)
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	$\overline{\text{TO}}$	$\overline{\text{PD}}$	—	—	--11 11--	--11 qq--

Legend: - = unimplemented read as '0', q - value depends on condition, shaded cells are not used by the WDT.
Note 1: Other (non power-up) resets include: external reset through MCLR and Watchdog Timer Reset.
2: This value will be as the device was programmed, or if unprogrammed, will read as all '1's.

TABLE 15-2: PIC17CXX INSTRUCTION SET

Mnemonic, Operands	Description	Cycles	16-bit Opcode		Status Affected	Notes
			MSb	LSb		
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f,d	ADD WREG to f	1	0000 111d ffff ffff	OV,C,DC,Z	
ADDWFC	f,d	ADD WREG and Carry bit to f	1	0001 000d ffff ffff	OV,C,DC,Z	
ANDWF	f,d	AND WREG with f	1	0000 101d ffff ffff	Z	
CLRF	f,s	Clear f, or Clear f and Clear WREG	1	0010 100s ffff ffff	None	3
COMF	f,d	Complement f	1	0001 001d ffff ffff	Z	
CPFSEQ	f	Compare f with WREG, skip if f = WREG	1 (2)	0011 0001 ffff ffff	None	6,8
CPFSGT	f	Compare f with WREG, skip if f > WREG	1 (2)	0011 0010 ffff ffff	None	2,6,8
CPFSLT	f	Compare f with WREG, skip if f < WREG	1 (2)	0011 0000 ffff ffff	None	2,6,8
DAW	f,s	Decimal Adjust WREG Register	1	0010 111s ffff ffff	C	3
DECF	f,d	Decrement f	1	0000 011d ffff ffff	OV,C,DC,Z	
DECFSZ	f,d	Decrement f, skip if 0	1 (2)	0001 011d ffff ffff	None	6,8
DCFSNZ	f,d	Decrement f, skip if not 0	1 (2)	0010 011d ffff ffff	None	6,8
INCF	f,d	Increment f	1	0001 010d ffff ffff	OV,C,DC,Z	
INCFSZ	f,d	Increment f, skip if 0	1 (2)	0001 111d ffff ffff	None	6,8
INFSNZ	f,d	Increment f, skip if not 0	1 (2)	0010 010d ffff ffff	None	6,8
IORWF	f,d	Inclusive OR WREG with f	1	0000 100d ffff ffff	Z	
MOVFP	f,p	Move f to p	1	011p pppp ffff ffff	None	
MOVPF	p,f	Move p to f	1	010p pppp ffff ffff	Z	
MOVWF	f	Move WREG to f	1	0000 0001 ffff ffff	None	
MULWF	f	Multiply WREG with f	1	0011 0100 ffff ffff	None	9
NEGW	f,s	Negate WREG	1	0010 110s ffff ffff	OV,C,DC,Z	1,3
NOP	—	No Operation	1	0000 0000 0000 0000	None	
RLCF	f,d	Rotate left f through Carry	1	0001 101d ffff ffff	C	
RLNCF	f,d	Rotate left f (no carry)	1	0010 001d ffff ffff	None	
RRCF	f,d	Rotate right f through Carry	1	0001 100d ffff ffff	C	
RRNCF	f,d	Rotate right f (no carry)	1	0010 000d ffff ffff	None	
SETF	f,s	Set f	1	0010 101s ffff ffff	None	3
SUBWF	f,d	Subtract WREG from f	1	0000 010d ffff ffff	OV,C,DC,Z	1
SUBWFB	f,d	Subtract WREG from f with Borrow	1	0000 001d ffff ffff	OV,C,DC,Z	1
SWAPF	f,d	Swap f	1	0001 110d ffff ffff	None	
TABLRD	t,i,f	Table Read	2 (3)	1010 10ti ffff ffff	None	7

Legend: Refer to Table 15-1 for opcode field descriptions.

Note 1: 2's Complement method.

- 2: Unsigned arithmetic.
- 3: If s = '1', only the file is affected: If s = '0', both the WREG register and the file are affected; If only the Working register (WREG) is required to be affected, then f = WREG must be specified.
- 4: During an LCALL, the contents of PCLATH are loaded into the MSB of the PC and kkkk kkkk is loaded into the LSB of the PC (PCL)
- 5: Multiple cycle instruction for EPROM programming when table pointer selects internal EPROM. The instruction is terminated by an interrupt event. When writing to external program memory, it is a two-cycle instruction.
- 6: Two-cycle instruction when condition is true, else single cycle instruction.
- 7: Two-cycle instruction except for TABLRD to PCL (program counter low byte) in which case it takes 3 cycles.
- 8: A "skip" means that instruction fetched during execution of current instruction is not executed, instead an NOP is executed.
- 9: These instructions are not available on the PIC17C42.

PIC17C4X

NOTES:

Applicable Devices	42	R42	42A	43	R43	44
--------------------	----	-----	-----	----	-----	----

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature					
		-40°C ≤ TA ≤ +40°C					
		Operating voltage VDD range as described in Section 19.1					
Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
		Internal Program Memory Programming Specs (Note 4)					
D110	VPP	Voltage on $\overline{\text{MCLR}}/\text{VPP}$ pin	12.75	–	13.25	V	Note 5
D111	VDDP	Supply voltage during programming	4.75	5.0	5.25	V	
D112	IPP	Current into $\overline{\text{MCLR}}/\text{VPP}$ pin	–	25 ‡	50 ‡	mA	
D113	IDDP	Supply current during programming	–	–	30 ‡	mA	
D114	TPROG	Programming pulse width	10	100	1000	µs	Terminated via internal/external interrupt or a reset

* 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 $\overline{\text{MCLR}}$ 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 $\overline{\text{MCLR}}/\text{VPP}$ 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.

Note: When using the Table Write for internal programming, the device temperature must be less than 40°C.

PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

19.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I²C specifications only)
4. Ts (I²C specifications only)

T		T	
F	Frequency	T	Time

Lowercase symbols (pp) and their meanings:

pp			
ad	Address/Data	ost	Oscillator Start-Up Timer
al	ALE	pwrt	Power-Up Timer
cc	Capture1 and Capture2	rb	PORTB
ck	CLKOUT or clock	rd	$\overline{RD}$
dt	Data in	rw	$\overline{RD}$ or $\overline{WR}$
in	INT pin	t0	T0CKI
io	I/O port	t123	TCLK12 and TCLK3
mc	$\overline{MCLR}$	wdt	Watchdog Timer
oe	$\overline{OE}$	wr	$\overline{WR}$
os	OSC1		

Uppercase symbols and their meanings:

S			
D	Driven	L	Low
E	Edge	P	Period
F	Fall	R	Rise
H	High	V	Valid
I	Invalid (Hi-impedance)	Z	Hi-impedance

PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

19.5 Timing Diagrams and Specifications

FIGURE 19-2: EXTERNAL CLOCK TIMING

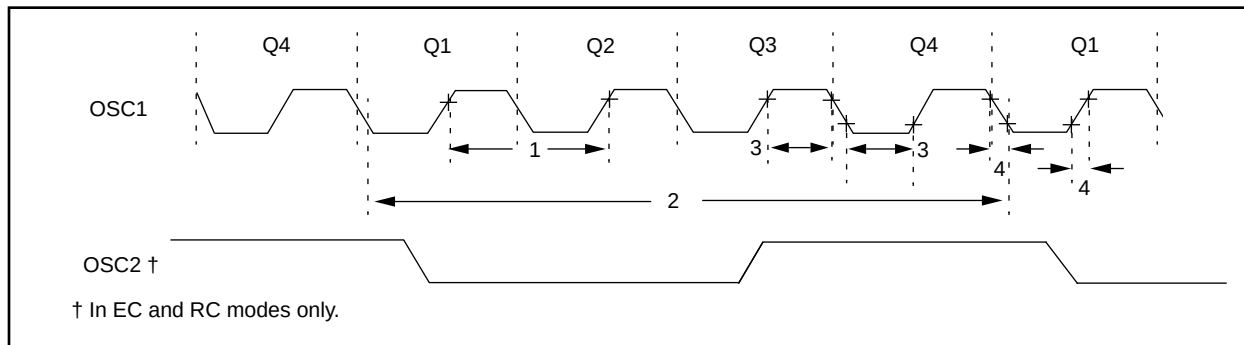


TABLE 19-2: EXTERNAL CLOCK TIMING REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	External CLKIN Frequency (Note 1)	DC	—	8	MHz	EC osc mode - 08 devices (8 MHz devices)
			DC	—	16	MHz	- 16 devices (16 MHz devices)
			DC	—	25	MHz	- 25 devices (25 MHz devices)
			DC	—	33	MHz	- 33 devices (33 MHz devices)
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			1	—	8	MHz	XT osc mode - 08 devices (8 MHz devices)
			1	—	16	MHz	- 16 devices (16 MHz devices)
			1	—	25	MHz	- 25 devices (25 MHz devices)
1	Tosc	External CLKIN Period (Note 1)	1	—	33	MHz	- 33 devices (33 MHz devices)
			DC	—	2	MHz	LF osc mode
			250	—	—	ns	RC osc mode
			125	—	1,000	ns	XT osc mode - 08 devices (8 MHz devices)
		Oscillator Period (Note 1)	62.5	—	1,000	ns	- 16 devices (16 MHz devices)
			40	—	1,000	ns	- 25 devices (25 MHz devices)
			30.3	—	1,000	ns	- 33 devices (33 MHz devices)
			500	—	—	ns	LF osc mode
2	Tcy	Instruction Cycle Time (Note 1)	121.2	4/Fosc	DC	ns	
3	TosL, TosH	Clock in (OSC1) high or low time	10 ‡	—	—	ns	EC oscillator
4	TosR, TosF	Clock in (OSC1) rise or fall time	—	—	5 ‡	ns	EC oscillator

† 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: Instruction cycle period (Tcy) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "max." cycle time limit is "DC" (no clock) for all devices.

PIC17C4X

NOTES:

PIC17C4X

NOTES:

PIC17C4X

E.6 PIC16C8X Family of Devices

	Clock			Memory			Peripherals		Features				
	Maximum Frequency of Operation (MHz)			Program Memory			Data EEPROM (bytes)	Timer Modules(s)		Interrupt Sources	I/O Pins	Voltage Range (Volts)	Packages
	Flash	EEPROM	ROM	Data Memory (bytes)	Data EEPROM (bytes)								
PIC16C84	10	—	1K	—	64	TMR0	4	13	2.0-6.0	18-pin DIP, SOIC			
PIC16F84 ⁽¹⁾	10	1K	—	—	64	TMR0	4	13	2.0-6.0	18-pin DIP, SOIC			
PIC16CR84 ⁽¹⁾	10	—	—	1K	64	TMR0	4	13	2.0-6.0	18-pin DIP, SOIC			
PIC16F83 ⁽¹⁾	10	512	—	—	64	TMR0	4	13	2.0-6.0	18-pin DIP, SOIC			
PIC16CR83 ⁽¹⁾	10	—	—	512	64	TMR0	4	13	2.0-6.0	18-pin DIP, SOIC			

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

All PIC16C8X family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local sales office for availability of these devices.

WDT	99, 103
Clearing the WDT	103
Normal Timer	103
Period	103
Programming Considerations	103
WDTPS0	99
WDTPS1	99
WREG	34

X

XORLW	141
XORWF	141

Z

Z	9, 36
Zero (Z)	9

LIST OF EXAMPLES

Example 3-1: Signed Math	9
Example 3-2: Instruction Pipeline Flow	14
Example 5-1: Saving STATUS and WREG in RAM	27
Example 6-1: Indirect Addressing	40
Example 7-1: Table Write	46
Example 7-2: Table Read	48
Example 8-1: 8 x 8 Multiply Routine	49
Example 8-2: 8 x 8 Signed Multiply Routine	49
Example 8-3: 16 x 16 Multiply Routine	50
Example 8-4: 16 x 16 Signed Multiply Routine	51
Example 9-1: Initializing PORTB	57
Example 9-2: Initializing PORTC	58
Example 9-3: Initializing PORTD	60
Example 9-4: Initializing PORTE	62
Example 9-5: Read Modify Write Instructions on an I/O Port	64
Example 11-1: 16-Bit Read	69
Example 11-2: 16-Bit Write	69
Example 12-1: Sequence to Read Capture Registers	78
Example 12-2: Writing to TMR3	80
Example 12-3: Reading from TMR3	80
Example 13-1: Calculating Baud Rate Error	86
Example F-1: PIC17C42 to Sleep	223

LIST OF FIGURES

Figure 3-1: PIC17C42 Block Diagram	10
Figure 3-2: PIC17CR42/42A/43/R43/44 Block Diagram	11
Figure 3-3: Clock/Instruction Cycle	14
Figure 4-1: Simplified Block Diagram of On-chip Reset Circuit	15
Figure 4-2: Time-Out Sequence on Power-Up (MCLR Tied to VDD)	17
Figure 4-3: Time-Out Sequence on Power-Up (MCLR NOT Tied to VDD)	17
Figure 4-4: Slow Rise Time (MCLR Tied to VDD)	17
Figure 4-5: Oscillator Start-Up Time	18
Figure 4-6: Using On-Chip POR	18
Figure 4-7: Brown-out Protection Circuit 1	18
Figure 4-8: PIC17C42 External Power-On Reset Circuit (For Slow VDD Power-Up)	18
Figure 4-9: Brown-out Protection Circuit 2	18
Figure 5-1: Interrupt Logic	21
Figure 5-2: INTSTA Register (Address: 07h, Unbanked)	22
Figure 5-3: PIE Register (Address: 17h, Bank 1)	23
Figure 5-4: PIR Register (Address: 16h, Bank 1)	24
Figure 5-5: INT Pin / T0CKI Pin Interrupt Timing	26
Figure 6-1: Program Memory Map and Stack	29
Figure 6-2: Memory Map in Different Modes	30
Figure 6-3: External Program Memory Access Waveforms	31
Figure 6-4: Typical External Program Memory Connection Diagram	31
Figure 6-5: PIC17C42 Register File Map	33
Figure 6-6: PIC17CR42/42A/43/R43/44 Register File Map	33
Figure 6-7: ALUSTA Register (Address: 04h, Unbanked)	36
Figure 6-8: CPUSTA Register (Address: 06h, Unbanked)	37
Figure 6-9: TOSTA Register (Address: 05h, Unbanked)	38
Figure 6-10: Indirect Addressing	39
Figure 6-11: Program Counter Operation	41