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	8MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	66
Program Memory Size	32KB (16K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	902 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17lc766t-08-pt

2.0 DEVICE VARIETIES

Each device has a variety of frequency ranges and packaging options. Depending on application and production requirements, the proper device option can be selected using the information in the PIC17C7XX Product Selection System section at the end of this data sheet. When placing orders, please use the "PIC17C7XX Product Identification System" at the back of this data sheet to specify the correct part number. When discussing the functionality of the device, memory technology and voltage range does not matter.

There are two memory type options. These are specified in the middle characters of the part number.

1. **C**, as in PIC17**C**756A. These devices have EPROM type memory.
2. **CR**, as in PIC17**CR**756A. These devices have ROM type memory.

All these devices operate over the standard voltage range. Devices are also offered which operate over an extended voltage range (and reduced frequency range). Table 2-1 shows all possible memory types and voltage range designators for a particular device. These designators are in **bold** typeface.

TABLE 2-1: DEVICE MEMORY VARIETIES

Memory Type	Voltage Range	
	Standard	**Extended**
EPROM	PIC17**C**XXX	PIC17**LC**XXX
ROM	PIC17**CR**XXX	PIC17**LCR**XXX

Note: Not all memory technologies are available for a particular device.

2.1 UV Erasable Devices

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

The UV erasable version can be erased and reprogrammed to any of the configuration modes. 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 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 (SQTP^{sm}) 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.

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.

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

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

Note: Presently, NO ROM versions of the PIC17C7XX devices are available.

6.2 Peripheral Interrupt Enable Register1 (PIE1) and Register2 (PIE2)

These registers contains the individual enable bits for the peripheral interrupts.

REGISTER 6-2: PIE1 REGISTER (ADDRESS: 17h, BANK 1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TX1IE	RC1IE
bit 7							bit 0

bit 7 **RBIE**: PORTB Interrupt-on-Change Enable bit
1 = Enable PORTB interrupt-on-change
0 = Disable PORTB interrupt-on-change

bit 6 **TMR3IE**: TMR3 Interrupt Enable bit
1 = Enable TMR3 interrupt
0 = Disable TMR3 interrupt

bit 5 **TMR2IE**: TMR2 Interrupt Enable bit
1 = Enable TMR2 interrupt
0 = Disable TMR2 interrupt

bit 4 **TMR1IE**: TMR1 Interrupt Enable bit
1 = Enable TMR1 interrupt
0 = Disable TMR1 interrupt

bit 3 **CA2IE**: Capture2 Interrupt Enable bit
1 = Enable Capture2 interrupt
0 = Disable Capture2 interrupt

bit 2 **CA1IE**: Capture1 Interrupt Enable bit
1 = Enable Capture1 interrupt
0 = Disable Capture1 interrupt

bit 1 **TX1IE**: USART1 Transmit Interrupt Enable bit
1 = Enable USART1 Transmit buffer empty interrupt
0 = Disable USART1 Transmit buffer empty interrupt

bit 0 **RC1IE**: USART1 Receive Interrupt Enable bit
1 = Enable USART1 Receive buffer full interrupt
0 = Disable USART1 Receive buffer full interrupt

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR Reset	’1’ = Bit is set	’0’ = Bit is cleared	x = Bit is unknown

8.0 TABLE READS AND TABLE WRITES

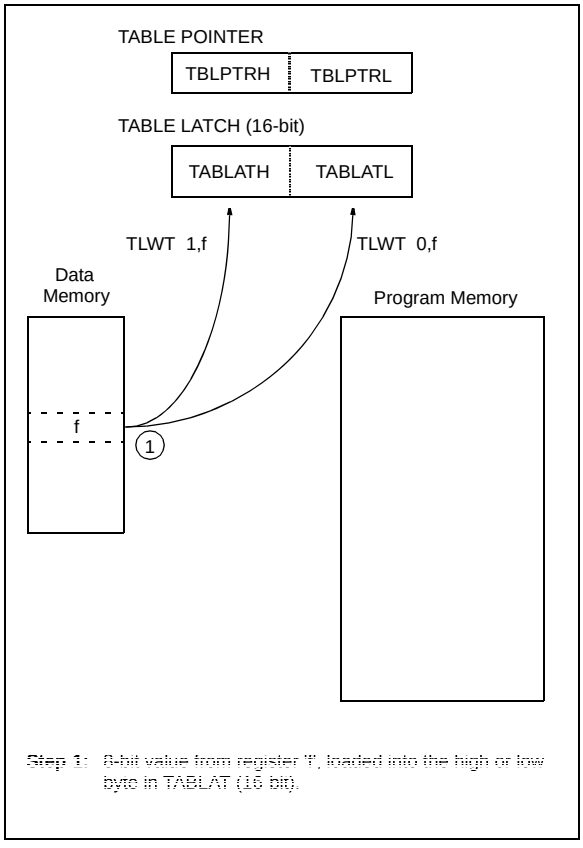
The PIC17C7XX has four instructions that allow the processor to move data from the data memory space to the program memory space, and vice versa. Since the program memory space is 16-bits wide and the data memory space is 8-bits wide, two operations are required to move 16-bit values to/from the data memory.

The `TLWT t,f` and `TABLWT t,i,f` instructions are used to write data from the data memory space to the program memory space. The `TLRD t,f` and `TABLRD t,i,f` instructions are used to write data from the program memory space to the data memory space.

The program memory can be internal or external. For the program memory access to be external, the device needs to be operating in Microprocessor or Extended Microcontroller mode.

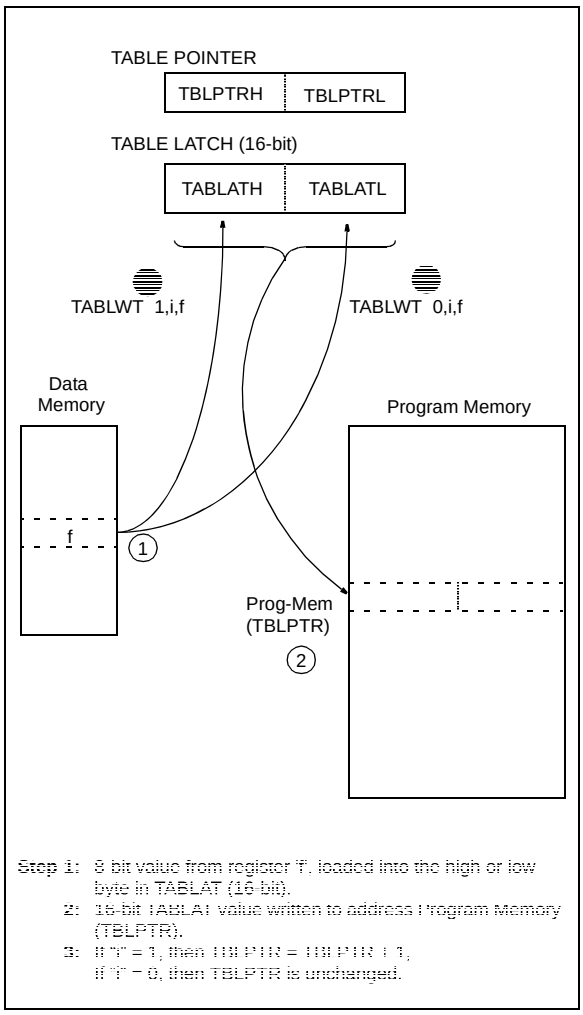
Figure 8-1 through Figure 8-4 show the operation of these four instructions. The steps show the sequence of operation.

FIGURE 8-1: `TLWT` INSTRUCTION OPERATION



Step 1: 8-bit value from register 'f', loaded into the high or low byte in TABLAT (16 bit).

FIGURE 8-2: `TABLWT` INSTRUCTION OPERATION



Step 1: 8 bit value from register 'f', loaded into the high or low byte in TABLAT (16 bit).
2: 16-bit TABLAT value written to address Program Memory (TBLPTR).
3: If "t" = 1, then TBLPTR = TBLPTR + 1,
if "t" = 0, then TBLPTR is unchanged.

10.6 PORTF and DDRF Registers

PORTF is an 8-bit wide bi-directional port. The corresponding data direction register is DDRF. A '1' in DDRF configures the corresponding port pin as an input. A '0' in the DDRF register configures the corresponding port pin as an output. Reading PORTF reads the status of the pins, whereas writing to PORTF will write to the respective port latch.

All eight bits of PORTF are multiplexed with 8 channels of the 10-bit A/D converter.

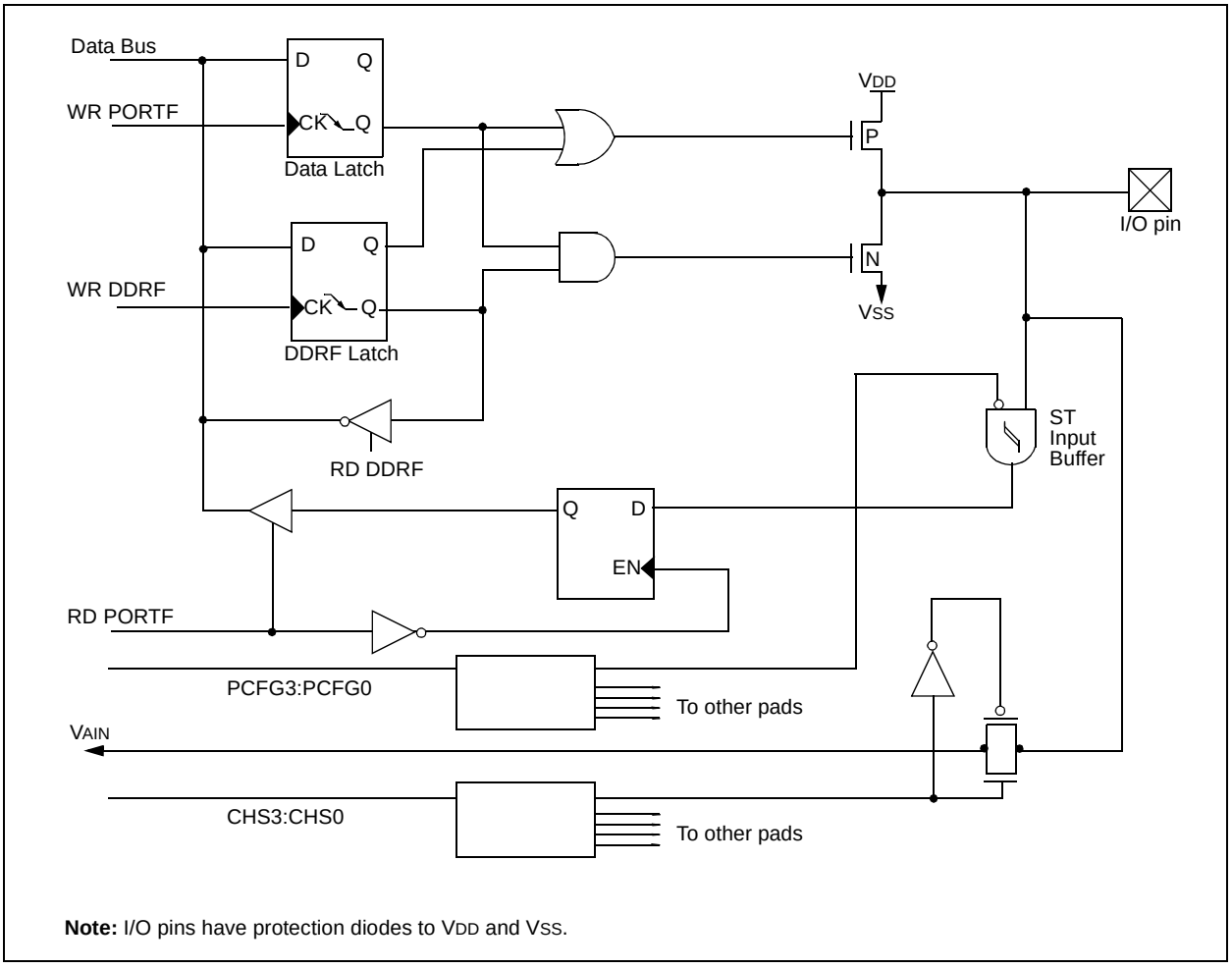
Upon RESET, the entire Port is automatically configured as analog inputs and must be configured in software to be a digital I/O.

Example 10-6 shows an instruction sequence to initialize PORTF. The Bank Select Register (BSR) must be selected to Bank 5 for the port to be initialized. The following example uses the `MOVLB` instruction to load the BSR register for bank selection.

EXAMPLE 10-6: INITIALIZING PORTF

```
MOVLB   5          ; Select Bank 5
MOVWF   0x0E       ; Configure PORTF as
MOVWF   ADCON1     ; Digital
CLRF    PORTF, F   ; Initialize PORTF data
                   ;   latches before
                   ;   the data direction
                   ;   register
MOVLW    0x03      ; Value used to init
                   ;   data direction
MOVWF    DDRF      ; Set RF<1:0> as inputs
                   ;   RF<7:2> as outputs
```

FIGURE 10-13: BLOCK DIAGRAM OF RF7:RF0



Note: I/O pins have protection diodes to V_{DD} and V_{SS}.

13.1.3.3 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 1TCY (one instruction cycle). This will cause jitter in the duty cycle as well as the period of the PWM output.

This jitter will be ±1TCY, unless the external clock is synchronized with the processor clock. Use of one of the PWM outputs as the clock source to the TCLK12 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).

13.1.3.4 Maximum Resolution/Frequency for External Clock Input

The use of an external clock for the PWM time base (Timer1 or Timer2) limits the PWM output to a maximum resolution of 8-bits. The PWxDCL<7:6> bits must be kept cleared. Use of any other value will distort the PWM output. All resolutions are supported when internal clock mode is selected. The maximum attainable frequency is also lower. This is a result of the timing requirements of an external clock input for a timer (see the Electrical Specification section). The maximum PWM frequency, when the timers clock source is the RB4/TCLK12 pin, is shown in Table 13-4 (Standard Resolution mode).

TABLE 13-5: REGISTERS/BITS ASSOCIATED WITH PWM

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{\text{MCLR}}$, WDT
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/$\overline{\text{PR3}}$	TMR3ON	TMR2ON	TMR1ON	0000 0000	0000 0000
16h, Bank 7	TCON3	—	CA4OVF	CA3OVF	CA4ED1	CA4ED0	CA3ED1	CA3ED0	PWM3ON	-000 0000	-000 0000
10h, Bank 2	TMR1	Timer1's Register								xxxx xxxx	uuuu uuuu
11h, Bank 2	TMR2	Timer2's Register								xxxx xxxx	uuuu uuuu
16h, Bank 1	PIR1	RBIF	TMR3IF	TMR2IF	TMR1IF	CA2IF	CA1IF	TX1IF	RC1IF	x000 0010	u000 0010
17h, Bank 1	PIE1	RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TX1IE	RC1IE	0000 0000	0000 0000
07h, Unbanked	INTSTA	PEIF	T0CKIF	T0IF	INTF	PEIE	T0CKIE	T0IE	INTE	0000 0000	0000 0000
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	$\overline{\text{TO}}$	$\overline{\text{PD}}$	POR	BOR	--11 11qq	--11 qquu
14h, Bank 2	PR1	Timer1 Period Register								xxxx xxxx	uuuu uuuu
15h, Bank 2	PR2	Timer2 Period Register								xxxx xxxx	uuuu uuuu
10h, Bank 3	PW1DCL	DC1	DC0	—	—	—	—	—	—	xx-- ----	uu-- ----
11h, Bank 3	PW2DCL	DC1	DC0	TM2PW2	—	—	—	—	—	xx0- ----	uu0- ----
10h, Bank 7	PW3DCL	DC1	DC0	TM2PW3	—	—	—	—	—	xx0- ----	uu0- ----
12h, Bank 3	PW1DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
13h, Bank 3	PW2DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
11h, Bank 7	PW3DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0', q = value depends on conditions.
Shaded cells are not used by PWM Module.

14.2 USART Asynchronous Mode

In this mode, the USART uses standard nonreturn-to-zero (NRZ) format (one START bit, eight or nine data bits, and one STOP bit). The most common data format is 8-bits. An on-chip dedicated 8-bit baud rate generator can be used to derive standard baud rate frequencies from the oscillator. The USART's transmitter and receiver are functionally independent but use the same data format and baud rate. The baud rate generator produces a clock x64 of the bit shift rate. Parity is not supported by the hardware, but can be implemented in software (and stored as the ninth data bit). Asynchronous mode is stopped during SLEEP.

The Asynchronous mode is selected by clearing the SYNC bit (TXSTA<4>).

The USART Asynchronous module consists of the following components:

- Baud Rate Generator
- Sampling Circuit
- Asynchronous Transmitter
- Asynchronous Receiver

14.2.1 USART ASYNCHRONOUS TRANSMITTER

The USART transmitter block diagram is shown in Figure 14-1. The heart of the transmitter is the transmit shift register (TSR). The shift register obtains its data from the read/write transmit buffer (TXREG). TXREG is loaded with data in software. The TSR is not loaded until the STOP bit has been transmitted from the previous load. As soon as the STOP bit is transmitted, the TSR is loaded with new data from the TXREG (if available). Once TXREG transfers the data to the TSR (occurs in one TCY at the end of the current BRG cycle), the TXREG is empty and an interrupt bit, TXIF, is set. This interrupt can be enabled/disabled by setting/clearing the TXIE bit. TXIF will be set, regardless of TXIE and cannot be reset in software. It will reset only when new data is loaded into TXREG. While TXIF indicates the status of the TXREG, the TRMT (TXSTA<1>) bit shows the status of the TSR.

TRMT is a read only bit which is set when the TSR is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR is empty.

Note:	The TSR is not mapped in data memory, so it is not available to the user.

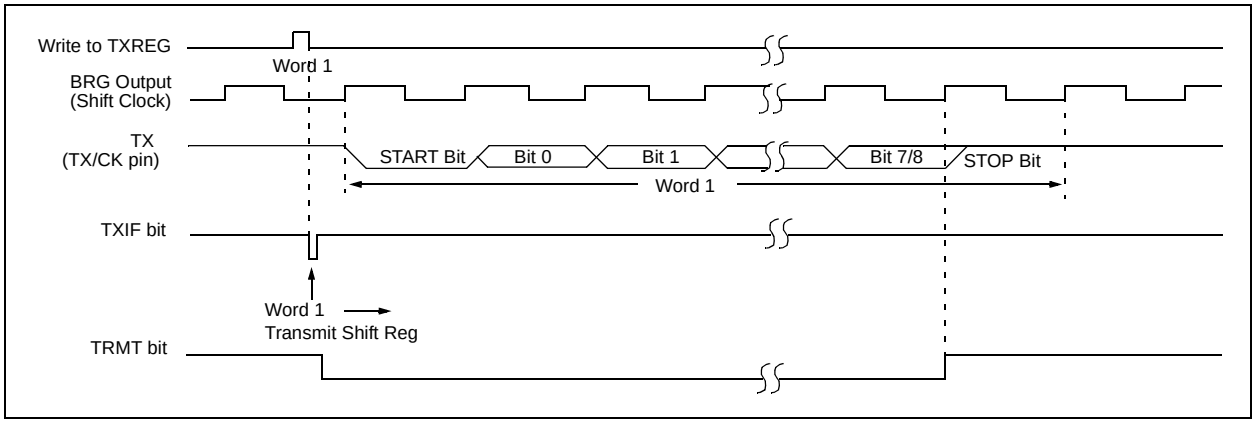
Transmission is enabled by setting the TXEN (TXSTA<5>) bit. The actual transmission will not occur until TXREG has been loaded with data and the baud rate generator (BRG) has produced a shift clock (Figure 14-3). The transmission can also be started by first loading TXREG and then setting TXEN. Normally, when transmission is first started, the TSR is empty, so a transfer to TXREG will result in an immediate transfer to TSR, resulting in an empty TXREG. A back-to-back transfer is thus possible (Figure 14-4). Clearing TXEN during a transmission will cause the transmission to be aborted. This will reset the transmitter and the TX/CK pin will revert to hi-impedance.

In order to select 9-bit transmission, the TX9 (TXSTA<6>) bit should be set and the ninth bit value should be written to TX9D (TXSTA<0>). The ninth bit value must be written before writing the 8-bit data to the TXREG. This is because a data write to TXREG can result in an immediate transfer of the data to the TSR (if the TSR is empty).

Steps to follow when setting up an Asynchronous Transmission:

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 TXIE bit.
4. If 9-bit transmission is desired, then set the TX9 bit.
5. If 9-bit transmission is selected, the ninth bit should be loaded in TX9D.
6. Load data to the TXREG register.
7. Enable the transmission by setting TXEN (starts transmission).

FIGURE 14-3: ASYNCHRONOUS MASTER TRANSMISSION



15.2.5 MASTER MODE

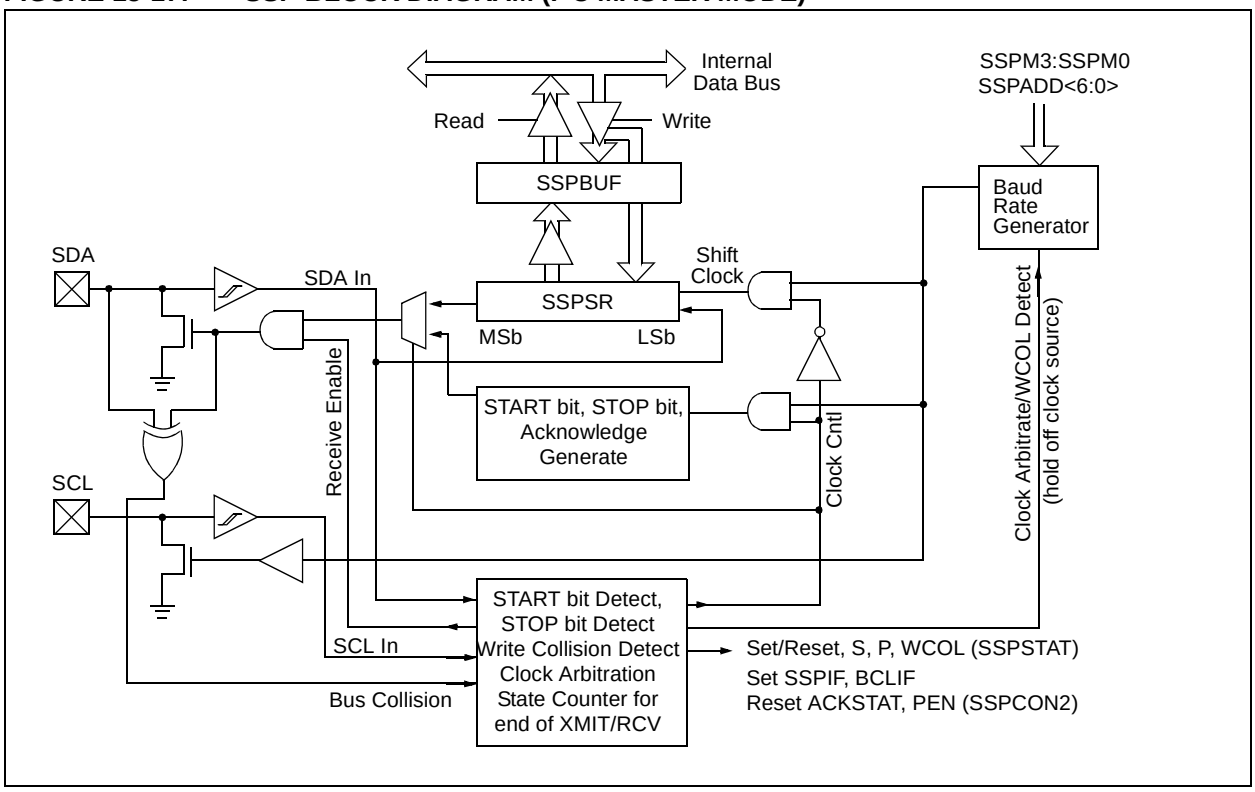
Master mode of operation is supported by interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a RESET, or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit is set, or the bus is idle, with both the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

In Master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received
- Acknowledge transmit
- Repeated Start

FIGURE 15-17: SSP BLOCK DIAGRAM (I^2C MASTER MODE)



A typical transmit sequence would go as follows:

a) The user generates a START Condition by setting the START enable bit (SEN) in SSPCON2.
b) SSPIF is set. The module will wait the required START time before any other operation takes place.
c) The user loads the SSPBUF with address to transmit.
d) Address is shifted out the SDA pin until all 8 bits are transmitted.
e) The MSSP Module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register (SSPCON2<6>).
f) The module generates an interrupt at the end of the ninth clock cycle by setting SSPIF.
g) The user loads the SSPBUF with eight bits of data.
h) DATA is shifted out the SDA pin until all 8 bits are transmitted.
i) The MSSP Module shifts in the ACK bit from the slave device, and writes its value into the SSPCON2 register (SSPCON2<6>).
j) The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
k) The user generates a STOP condition by setting the STOP enable bit PEN in SSPCON2.
l) Interrupt is generated once the STOP condition is complete.

15.2.8 BAUD RATE GENERATOR

In I^2C Master mode, the reload value for the BRG is located in the lower 7 bits of the SSPADD register (Figure 15-18). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload has taken place. The BRG count is decremented twice per instruction cycle (TCY), on the Q2 and Q4 clock.

In I^2C Master mode, the BRG is reloaded automatically. If Clock Arbitration is taking place, for instance, the BRG will be reloaded when the SCL pin is sampled high (Figure 15-19).

FIGURE 15-18: BAUD RATE GENERATOR BLOCK DIAGRAM

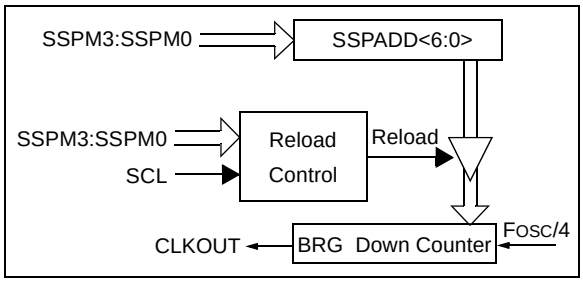
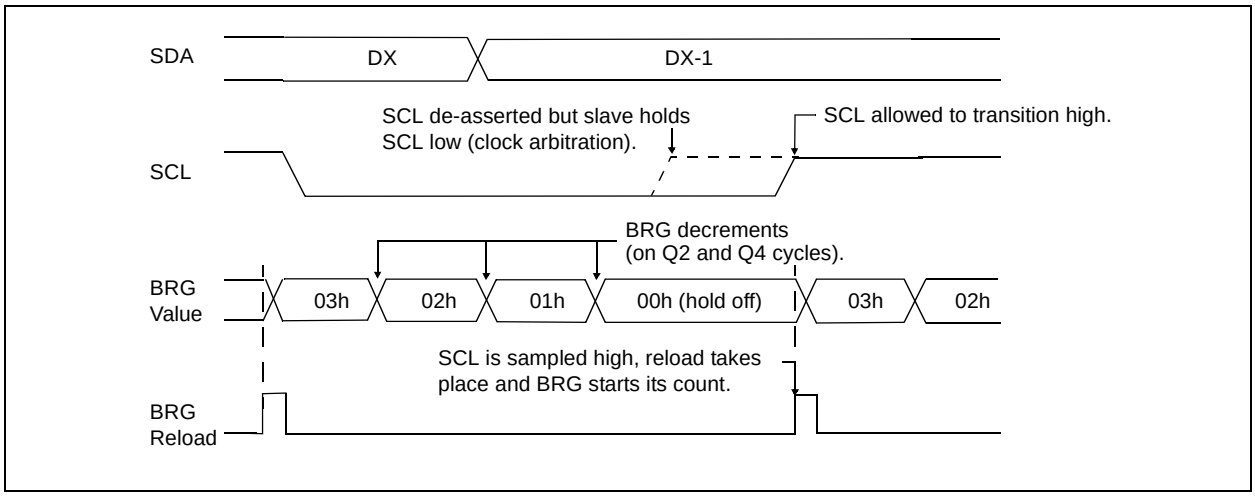


FIGURE 15-19: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



ADDWFC **ADD WREG and Carry bit to f**

Syntax: [*label*] ADDWFC f,d

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

Operation: (WREG) + (f) + C → (dest)

Status Affected: OV, C, DC, Z

Encoding:

0001	000d	ffff	ffff

Description: Add WREG, the Carry Flag and data memory location 'f'. If 'd' is 0, the result is placed in WREG. If 'd' is 1, the result is placed in data memory location 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `ADDWFC REG 0`

Before Instruction
Carry bit = 1
REG = 0x02
WREG = 0x4D

After Instruction
Carry bit = 0
REG = 0x02
WREG = 0x50

ANDLW **And Literal with WREG**

Syntax: [*label*] ANDLW k

Operands: $0 \le k \le 255$

Operation: (WREG) .AND. (k) → (WREG)

Status Affected: Z

Encoding:

1011	0101	kkkk	kkkk

Description: The contents of WREG are AND'ed with the 8-bit literal 'k'. The result is placed in WREG.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `ANDLW 0x5F`

Before Instruction
WREG = 0xA3

After Instruction
WREG = 0x03

CLRWDT **Clear Watchdog Timer**

Syntax: [*label*] CLRWDT

Operands: None

Operation: 00h → WDT
0 → $\overline{\text{WDT}}$ postscaler,
1 → $\overline{\text{TO}}$
1 → $\overline{\text{PD}}$

Status Affected: $\overline{\text{TO}}$, $\overline{\text{PD}}$

Encoding:

0000	0000	0000	0100

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the postscaler of the WDT. Status bits $\overline{\text{TO}}$ and $\overline{\text{PD}}$ are set.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	No operation

Example: CLRWDT

Before Instruction
WDT counter = ?

After Instruction
WDT counter = 0x00
$\overline{\text{WDT}}$ Postscaler = 0
$\overline{\text{TO}}$ = 1
$\overline{\text{PD}}$ = 1

COMF **Complement f**

Syntax: [*label*] COMF f,d

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

Operation: $(\bar{f}) \rightarrow$ (dest)

Status Affected: Z

Encoding:

0001	001d	ffff	ffff

Description: The contents of register 'f' are complemented. 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'	Process Data	Write to destination

Example: COMF REG1,0

Before Instruction
REG1 = 0x13

After Instruction
REG1 = 0x13
WREG = 0xEC

SWAPF **Swap f**

Syntax: [*label*] SWAPF f,d

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

Operation: f<3:0> → dest<7:4>;
f<7:4> → dest<3:0>

Status Affected: None

Encoding:

0001	110d	ffff	ffff

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0, the result is placed in WREG. If 'd' is 1, the result is placed in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `SWAPF REG, 0`

Before Instruction
REG = 0x53

After Instruction
REG = 0x35

TABLRD **Table Read**

Syntax: [*label*] TABLRD t,i,f

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

Operation: If t = 1,
TBLATH → f;
If t = 0,
TBLATL → f;
Prog Mem (TBLPTR) → TBLAT;
If i = 1,
TBLPTR + 1 → TBLPTR
If i = 0,
TBLPTR is unchanged

Status Affected: None

Encoding:

1010	10ti	ffff	ffff

Description:
1. A byte of the table latch (TBLAT) is moved to register file 'f'.
   If t = 1: the high byte is moved;
   If t = 0: the low byte is moved.
2. Then, the contents of the program memory location pointed to by the 16-bit Table Pointer (TBLPTR) are loaded into the 16-bit Table Latch (TBLAT).
3. If i = 1: TBLPTR is incremented;
   If i = 0: TBLPTR is not incremented.

Words: 1

Cycles: 2 (3-cycle if f = PCL)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register TBLATH or TBLATL	Process Data	Write register 'f'
No operation	No operation (Table Pointer on Address bus)	No operation	No operation ($\overline{OE}$ goes low)

TABLWT **Table Write**

Example1: `TABLWT 1, 1, REG`

Before Instruction

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

After Instruction (table write completion)

REG	=	0x53
TBLATH	=	0x53
TBLATL	=	0x55
TBLPTR	=	0xA357
MEMORY(TBLPTR - 1)	=	0x5355

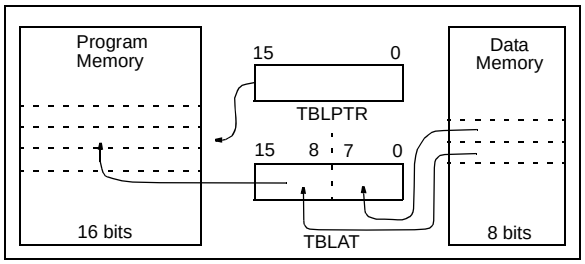
Example 2: `TABLWT 0, 0, REG`

Before Instruction

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

After Instruction (table write completion)

REG	=	0x53
TBLATH	=	0xAA
TBLATL	=	0x53
TBLPTR	=	0xA356
MEMORY(TBLPTR)	=	0xAA53



TLRD **Table Latch Read**

Syntax: [*label*] TLRD t,f

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

Operation: If t = 0,
TBLATL → f;
If t = 1,
TBLATH → f

Status Affected: None

Encoding:

1010	00tx	ffff	ffff

Description: Read data from 16-bit table latch (TBLAT) into file register 'f'. Table Latch is unaffected.
If t = 1; high byte is read
If t = 0; low byte is read
This instruction is used in conjunction with `TABLRD` to transfer data from program memory to data memory.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `TLRD t, RAM`

Before Instruction

t	=	0	
RAM	=	?	
TBLAT	=	0x00AF	(TBLATH = 0x00) (TBLATL = 0xAF)

After Instruction

RAM	=	0xAF	
TBLAT	=	0x00AF	(TBLATH = 0x00) (TBLATL = 0xAF)

Before Instruction

t	=	1	
RAM	=	?	
TBLAT	=	0x00AF	(TBLATH = 0x00) (TBLATL = 0xAF)

After Instruction

RAM	=	0x00	
TBLAT	=	0x00AF	(TBLATH = 0x00) (TBLATL = 0xAF)

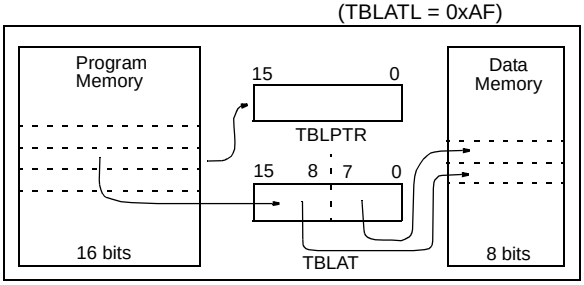


FIGURE 20-3: PIC17LC7XX-08 VOLTAGE-FREQUENCY GRAPH

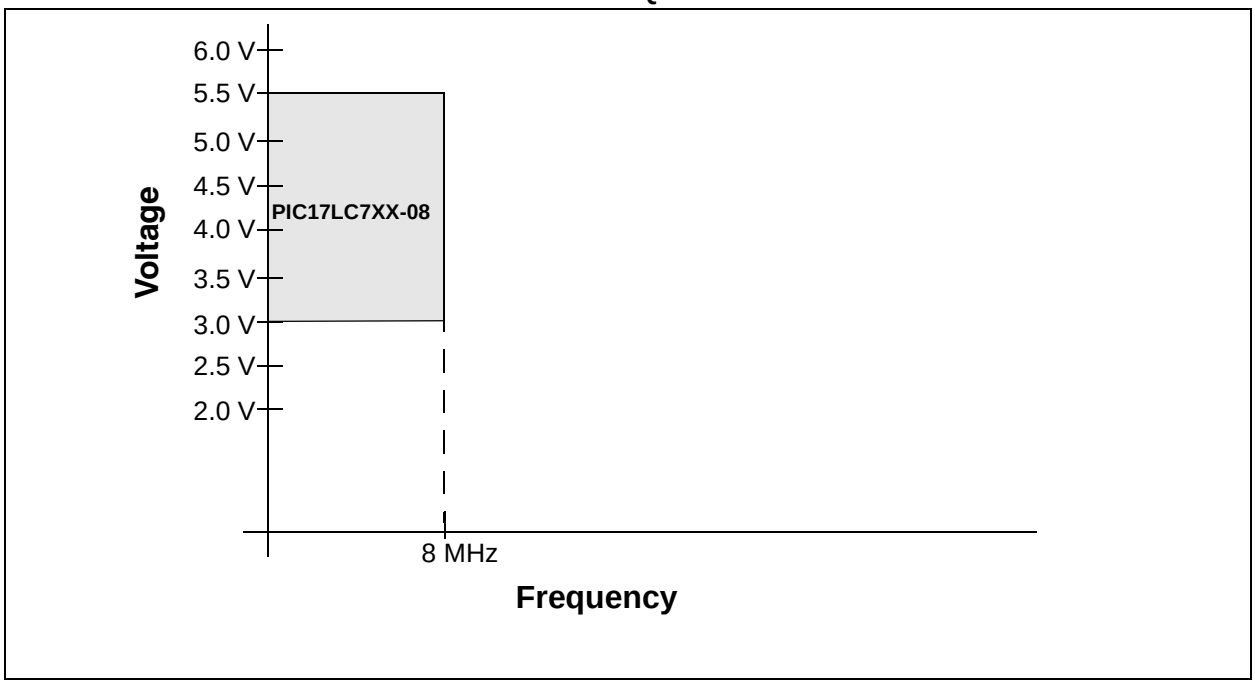
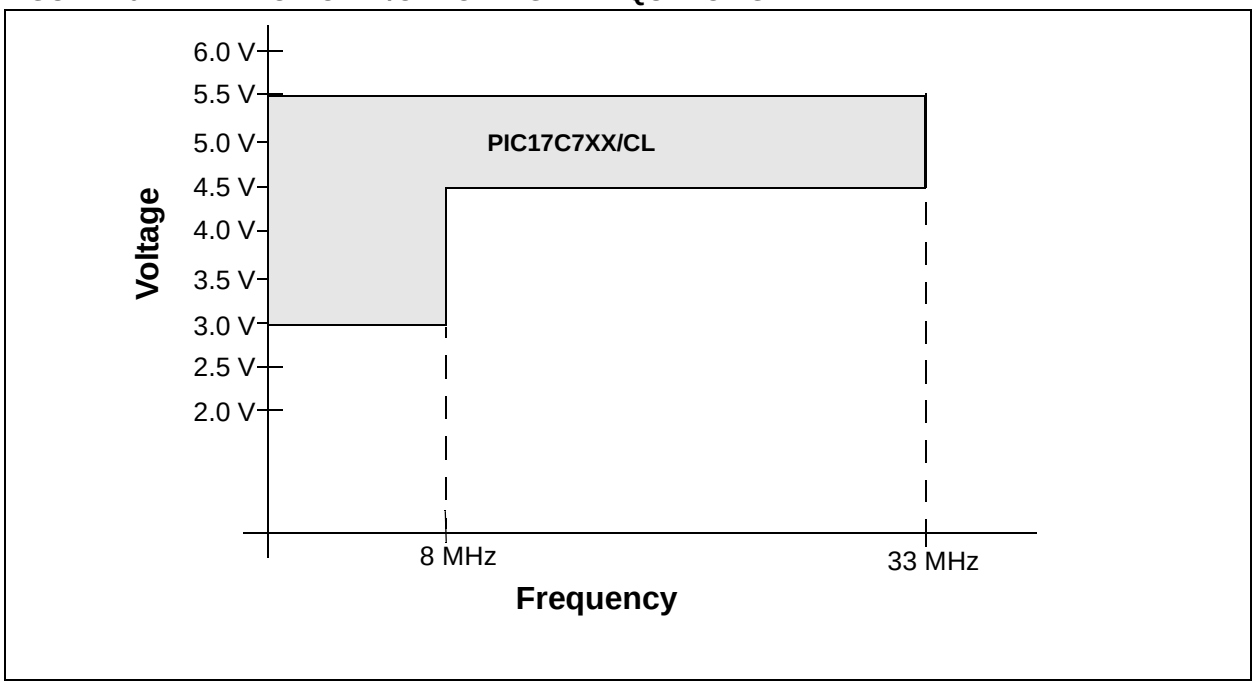


FIGURE 20-4: PIC17C7XX/CL VOLTAGE-FREQUENCY GRAPH



20.2 DC Characteristics: PIC17C7XX-16 (Commercial, Industrial, Extended) PIC17C7XX-33 (Commercial, Industrial, Extended) PIC17LC7XX-08 (Commercial, Industrial)

DC CHARACTERISTICS			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C for extended -40°C ≤ TA ≤ +85°C for industrial 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in Section 20.1				
Param. No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
		Input Low Voltage					
	VIL	I/O ports					
D030		with TTL buffer **(Note 6)**	Vss	–	0.8	V	4.5V ≤ VDD ≤ 5.5V
			Vss	–	0.2VDD	V	3.0V ≤ VDD ≤ 4.5V
D031		with Schmitt Trigger buffer					
		RA2, RA3	Vss	–	0.3VDD	V	I^2C compliant
		All others	Vss	–	0.2VDD	V	
D032		$\overline{MCLR}$, OSC1 (in EC and RC mode)	Vss	–	0.2VDD	V	**(Note 1)**
D033		OSC1 (in XT, and LF mode)	–	0.5VDD	–	V	
		Input High Voltage					
	VIH	I/O ports					
D040		with TTL buffer **(Note 6)**	2.0	–	VDD	V	4.5V ≤ VDD ≤ 5.5V
			1 + 0.2VDD	–	VDD	V	3.0V ≤ VDD ≤ 4.5V
D041		with Schmitt Trigger buffer					
		RA2, RA3	0.7VDD	–	VDD	V	I^2C compliant
		All others	0.8VDD	–	VDD	V	
D042		$\overline{MCLR}$	0.8VDD	–	VDD	V	**(Note 1)**
D043		OSC1 (XT, and LF mode)	–	0.5VDD	–	V	
D050	VHYS	**Hysteresis of Schmitt Trigger Inputs**	0.15VDD	–	–	V	

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC17CXXX 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. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by 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: PIC17C7XX Programming Specifications (Literature number DS TBD).

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.

20.3 Timing Parameter Symbology

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			
F	Frequency	T	Time

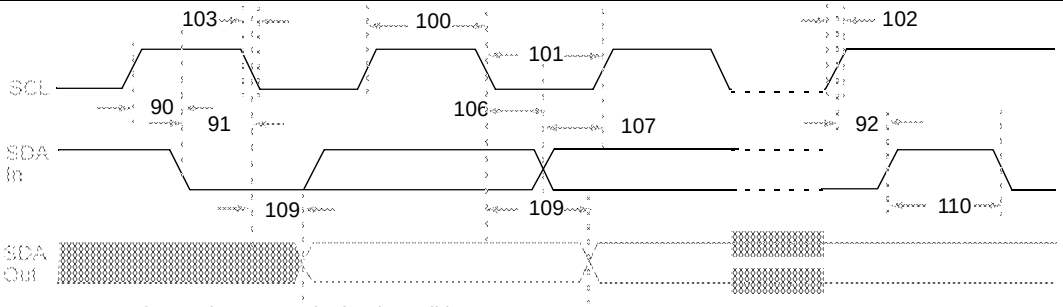
Lowercase symbols (pp) and their meanings:

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

FIGURE 20-18: I²C BUS DATA TIMING



Note: Refer to Figure 20-5 for load conditions.

TABLE 20-13: I²C BUS DATA REQUIREMENTS

Param No.	Sym	Characteristic		Min	Max	Units	Conditions
100	Thigh	Clock high time	100 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			400 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	ms	
101	Tlow	Clock low time	100 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			400 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	ms	
102	Tr	SDA and SCL rise time	100 kHz mode	—	1000	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1Cb	300	ns	
			1 MHz mode(1)	—	300	ns	
103	Tf	SDA and SCL fall time	100 kHz mode	—	300	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1Cb	300	ns	
			1 MHz mode(1)	—	10	ns	
90	Tsu:sta	START condition setup time	100 kHz mode	2(Tosc)(BRG + 1)	—	ms	Only relevant for Repeated Start condition
			400 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	ms	
91	Thd:sta	START condition hold time	100 kHz mode	2(Tosc)(BRG + 1)	—	ms	After this period, the first clock pulse is generated
			400 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	ms	
106	Thd:dat	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	ms	
			1 MHz mode(1)	0	—	ns	
107	Tsu:dat	Data input setup time	100 kHz mode	250	—	ns	**(Note 2)**
			400 kHz mode	100	—	ns	
			1 MHz mode(1)	100	—	ns	
92	Tsu:sto	STOP condition setup time	100 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			400 kHz mode	2(Tosc)(BRG + 1)	—	ms	
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	ms	
109	Taa	Output valid from clock	100 kHz mode	—	3500	ns	
			400 kHz mode	—	1000	ns	
			1 MHz mode(1)	—	400	ns	

Note 1: Maximum pin capacitance = 10 pF for all I²C pins.

2: A fast mode (400 KHz) I²C bus device can be used in a standard mode I²C bus system, but the parameter # 107 ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line. Parameter #102 + #107 = 1000 + 250 = 1250 ns (for 100 kHz mode) before the SCL line is released.

3: C_b is specified to be from 10-400pF. The minimum specifications are characterized with C_b=10pF. The rise time spec (t_r) is characterized with R_p=R_p min. The minimum fall time specification (t_f) is characterized with C_b=10pF,and R_p=R_p max. These are only valid for fast mode operation (V_{DD}=4.5-5.5V) and where the SPM bit (SSPSTAT<7>) =1.)

4: Max specifications for these parameters are valid for falling edge only. Specs are characterized with R_p=R_p min and C_b=400pF for standard mode, 200pF for fast mode, and 10pF for 1MHz mode.

FIGURE 20-24: MEMORY INTERFACE WRITE TIMING

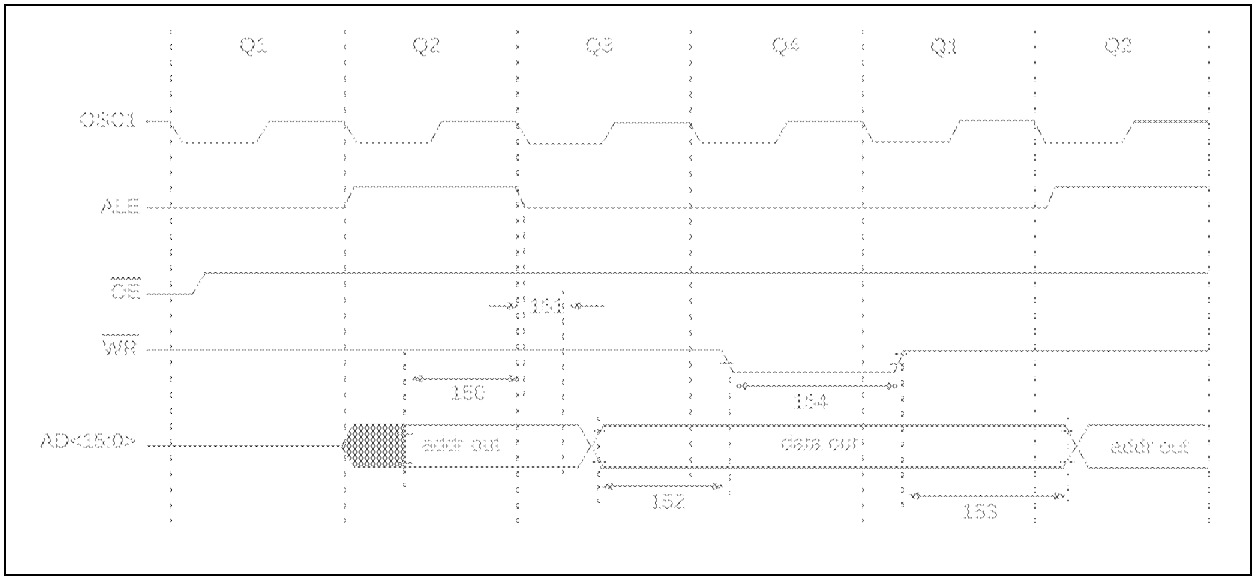


TABLE 20-20: MEMORY INTERFACE WRITE REQUIREMENTS

Param. No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
150	TadV2alL	AD<15:0> (address) valid to ALE↓ (address setup time)	PIC17**C**XXX	0.25TCY - 10	—	—	ns	
			PIC17**LC**XXX	0.25TCY - 10	—	—		
151	TalL2adI	ALE↓ to address out invalid (address hold time)	PIC17**C**XXX	0	—	—	ns	
			PIC17**LC**XXX	0	—	—		
152	TadV2wrL	Data out valid to $\overline{WR}$↓ (data setup time)	PIC17**C**XXX	0.25TCY - 40	—	—	ns	
			PIC17**LC**XXX	0.25TCY - 40	—	—		
153	TwrH2adI	$\overline{WR}$↑ to data out invalid (data hold time)	PIC17**C**XXX	—	0.25TCY	—	ns	
			PIC17**LC**XXX	—	0.25TCY	—		
154	TwrL	$\overline{WR}$ pulse width	PIC17**C**XXX	—	0.25TCY	—	ns	
			PIC17**LC**XXX	—	0.25TCY	—		

† Data in “Typ” column is at 5V, 25°C unless otherwise stated.

FIGURE 21-4: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

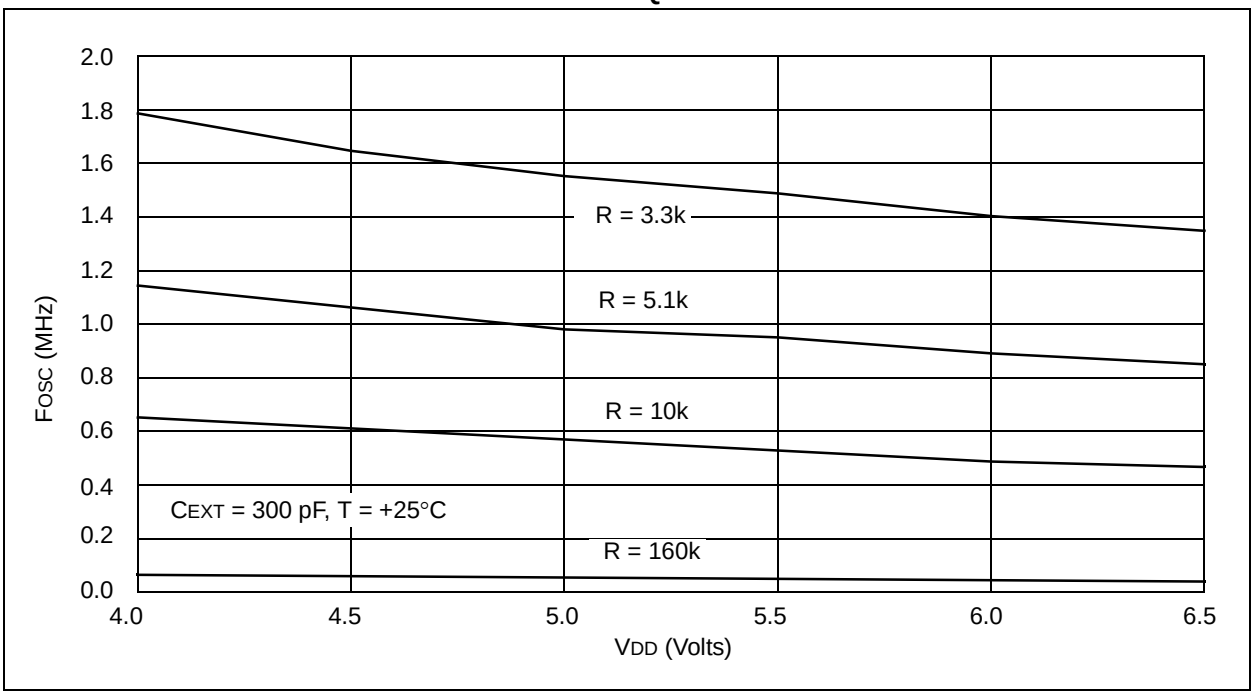


TABLE 21-2: RC OSCILLATOR FREQUENCIES

CEXT	REXT	Average FOSC @ 5V, +25°C	
22 pF	10k	3.33 MHz	± 12%
	100k	353 kHz	± 13%
100 pF	3.3k	3.54 MHz	± 10%
	5.1k	2.43 MHz	± 14%
	10k	1.30 MHz	± 17%
	100k	129 kHz	± 10%
300 pF	3.3k	1.54 MHz	± 14%
	5.1k	980 kHz	± 12%
	10k	564 kHz	± 16%
	160k	35 kHz	± 18%

FIGURE 21-11: TYPICAL AND MAXIMUM IPD vs. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED, -40°C to +125°C)

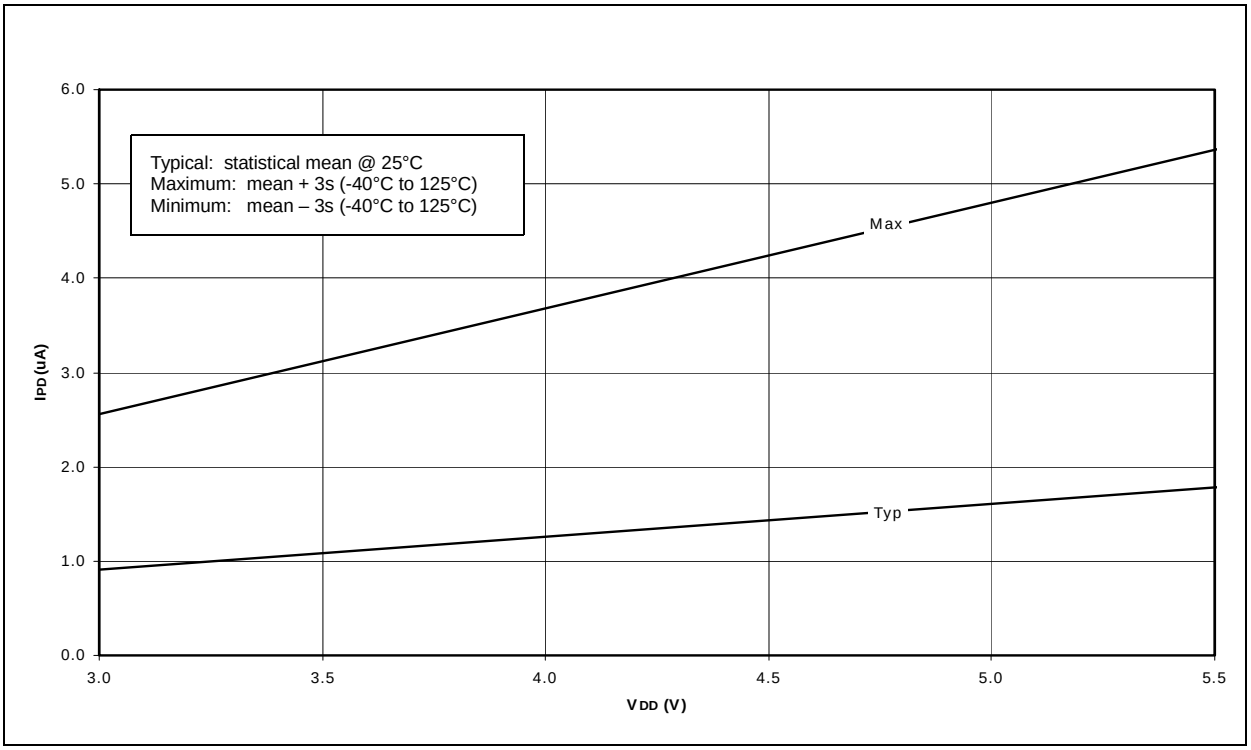


FIGURE 21-12: TYPICAL AND MAXIMUM IPD vs. VDD (SLEEP MODE, BOR ENABLED, -40°C to +125°C)