Microchip Technology - PIC17C762-16I/PT Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	16MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	66
Program Memory Size	16KB (8K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	678 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°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/pic17c762-16i-pt

TABLE 3-1: PINOUT DESCRIPTIONS

Name	PIC17C75X			PIC17C76X		I/O/P Type	Buffer Type	Description
	DIP No.	PLCC No.	TQFP No.	PLCC No.	QFP No.			
OSC1/CLKIN	47	50	39	62	49	I	ST	Oscillator input in Crystal/Resonator or RC Oscillator mode. External clock input in External Clock mode.
OSC2/CLKOUT	48	51	40	63	50	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 (FOSC/4) of OSC1 and denotes the instruction cycle rate.
$\overline{MCLR}$/VPP	15	16	7	20	9	I/P	ST	Master clear (RESET) input or Programming Voltage (VPP) input. This is the active low RESET input to the device.
								PORTA pins have individual differentiations that are listed in the following descriptions:
RA0/INT	56	60	48	72	58	I	ST	RA0 can also be selected as an external interrupt input. Interrupt can be configured to be on positive or negative edge. Input only pin.
RA1/T0CKI	41	44	33	56	43	I	ST	RA1 can also be selected as an external interrupt input and the interrupt can be configured to be on positive or negative edge. RA1 can also be selected to be the clock input to the Timer0 timer/counter. Input only pin.
RA2/$\overline{SS}$/SCL	42	45	34	57	44	I/O(2)	ST	RA2 can also be used as the slave select input for the SPI or the clock input for the I^2C bus. High voltage, high current, open drain port pin.
RA3/SDI/SDA	43	46	35	58	45	I/O(2)	ST	RA3 can also be used as the data input for the SPI or the data for the I^2C bus. High voltage, high current, open drain port pin.
RA4/RX1/DT1	40	43	32	51	38	I/O(1)	ST	RA4 can also be selected as the USART1 (SCI) Asynchronous Receive or USART1 (SCI) Synchronous Data. Output available from USART only.
RA5/TX1/CK1	39	42	31	50	37	I/O(1)	ST	RA5 can also be selected as the USART1 (SCI) Asynchronous Transmit or USART1 (SCI) Synchronous Clock. Output available from USART only.
								PORTB is a bi-directional I/O Port with software configurable weak pull-ups.
RB0/CAP1	55	59	47	71	57	I/O	ST	RB0 can also be the Capture1 input pin.
RB1/CAP2	54	58	46	70	56	I/O	ST	RB1 can also be the Capture2 input pin.
RB2/PWM1	50	54	42	66	52	I/O	ST	RB2 can also be the PWM1 output pin.
RB3/PWM2	53	57	45	69	55	I/O	ST	RB3 can also be the PWM2 output pin.
RB4/TCLK12	52	56	44	68	54	I/O	ST	RB4 can also be the external clock input to Timer1 and Timer2.
RB5/TCLK3	51	55	43	67	53	I/O	ST	RB5 can also be the external clock input to Timer3.
RB6/SCK	44	47	36	59	46	I/O	ST	RB6 can also be used as the master/slave clock for the SPI.
RB7/SDO	45	48	37	60	47	I/O	ST	RB7 can also be used as the data output for the SPI.

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

Note 1: The output is only available by the peripheral operation.
2: Open drain input/output pin. Pin forced to input upon any device RESET.

EXAMPLE 6-2: SAVING STATUS AND WREG IN RAM (NESTED)

```
; The addresses that are used to store the CPUSTA and WREG values must be in the data memory
; address range of 1Ah - 1Fh. Up to 6 locations can be saved and restored using the MOVFP
; instruction. This instruction neither affects the status bits, nor corrupts the WREG register.
; This routine uses the FRS0, so it controls the FS1 and FS0 bits in the ALUSTA register.
;
Nobank_FSR      EQU    0x40
Bank_FSR        EQU    0x41
ALU_Temp        EQU    0x42
WREG_TEMP       EQU    0x43
BSR_S1          EQU    0x01A        ; 1st location to save BSR
BSR_S2          EQU    0x01B        ; 2nd location to save BSR (Label Not used in program)
BSR_S3          EQU    0x01C        ; 3rd location to save BSR (Label Not used in program)
BSR_S4          EQU    0x01D        ; 4th location to save BSR (Label Not used in program)
BSR_S5          EQU    0x01E        ; 5th location to save BSR (Label Not used in program)
BSR_S6          EQU    0x01F        ; 6th location to save BSR (Label Not used in program)
;
INITIALIZATION                      ;
        CALL    CLEAR_RAM           ; Must Clear all Data RAM
;
INIT_POINTERS                       ; Must Initialize the pointers for POP and PUSH
        CLRF    BSR, F              ; Set All banks to 0
        CLRF    ALUSTA, F           ; FSR0 post increment
        BSF     ALUSTA, FS1
        CLRF    WREG, F             ; Clear WREG
        MOVLW   BSR_S1              ; Load FSR0 with 1st address to save BSR
        MOVWF   FSR0
        MOVWF   Nobank_FSR
        MOVLW   0x20
        MOVWF   Bank_FSR
        :
        :                           ; Your code
        :
        :                           ; At Interrupt Vector Address
PUSH    BSF     ALUSTA, FS0         ; FSR0 has auto-increment, does not affect status bits
        BCF     ALUSTA, FS1         ; does not affect status bits
        MOVFP   BSR, INDF0          ; No Status bits are affected
        CLRF    BSR, F              ; Peripheral and Data RAM Bank 0 No Status bits are affected
        MOVPF   ALUSTA, ALU_Temp    ;
        MOVPF   FSR0, Nobank_FSR    ; Save the FSR for BSR values
        MOVPF   WREG, WREG_TEMP     ;
        MOVFP   Bank_FSR, FSR0      ; Restore FSR value for other values
        MOVFP   ALU_Temp, INDF0     ; Push ALUSTA value
        MOVFP   WREG_TEMP, INDF0    ; Push WREG value
        MOVFP   PCLATH, INDF0       ; Push PCLATH value
        MOVPF   FSR0, Bank_FSR      ; Restore FSR value for other values
        MOVFP   Nobank_FSR, FSR0    ;
        ;
        :                           ; Interrupt Service Routine (ISR) code
        ;
POP     CLRF    ALUSTA, F           ; FSR0 has auto-decrement, does not affect status bits
        MOVFP   Bank_FSR, FSR0      ; Restore FSR value for other values
        DECF    FSR0, F             ;
        MOVFP   INDF0, PCLATH       ; Pop PCLATH value
        MOVFP   INDF0, WREG         ; Pop WREG value
        BSF     ALUSTA, FS1         ; FSR0 does not change
        MOVPF   INDF0, ALU_Temp     ; Pop ALUSTA value
        MOVPF   FSR0, Bank_FSR      ; Restore FSR value for other values
        DECF    Nobank_FSR, F       ;
        MOVFP   Nobank_FSR, FSR0    ; Save the FSR for BSR values
        MOVFP   ALU_Temp, ALUSTA    ;
        MOVFP   INDF0, BSR          ; No Status bits are affected
;
        RETFIE                      ; Return from interrupt (enable interrupts)
```

8.0 TABLE READS AND TABLE WRITES

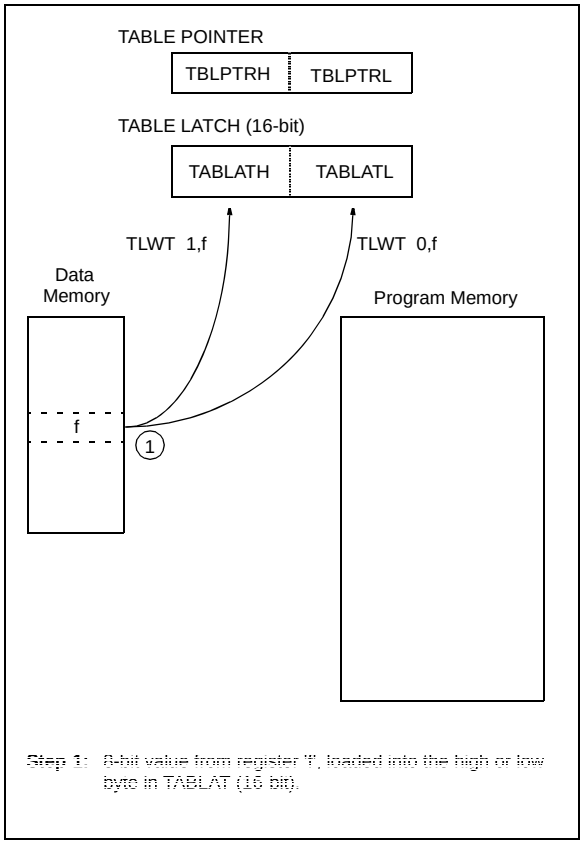
The PIC17C7XX has four instructions that allow the processor to move data from the data memory space to the program memory space, and vice versa. Since the program memory space is 16-bits wide and the data memory space is 8-bits wide, two operations are required to move 16-bit values to/from the data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

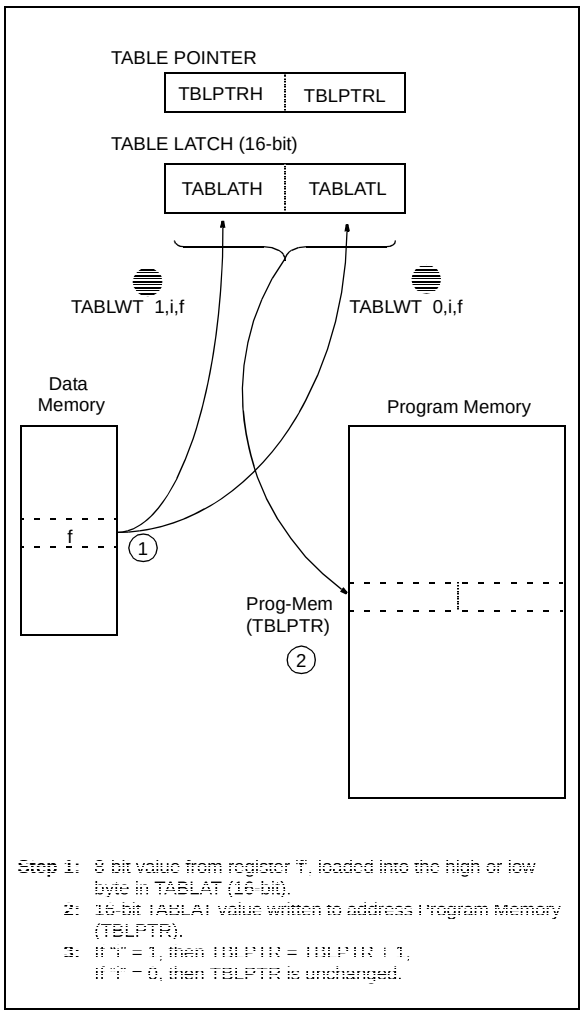
Figure 8-1 through Figure 8-4 show the operation of these four instructions. The steps show the sequence of operation.

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.

14.3 USART Synchronous Master Mode

In Master Synchronous mode, the data is transmitted in a half-duplex manner; i.e., transmission and reception do not occur at the same time: when transmitting data, the reception is inhibited and vice versa. The synchronous mode is entered by setting the SYNC (TXSTA<4>) bit. In addition, the SPEN (RCSTA<7>) bit is set in order to configure the I/O pins to CK (clock) and DT (data) lines, respectively. The Master mode indicates that the processor transmits the master clock on the CK line. The Master mode is entered by setting the CSRC (TXSTA<7>) bit.

14.3.1 USART SYNCHRONOUS MASTER TRANSMISSION

The USART transmitter block diagram is shown in Figure 14-1. The heart of the transmitter is the transmit (serial) 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 last bit has been transmitted from the previous load. As soon as the last bit is transmitted, the TSR is loaded with new data from TXREG (if available). Once TXREG transfers the data to the TSR (occurs in one T_{CY} at the end of the current BRG cycle), TXREG is empty and the TXIF bit is set. This interrupt can be enabled/disabled by setting/clearing the TXIE bit. TXIF will be set regardless of the state of bit TXIE and cannot be cleared in software. It will reset only when new data is loaded into TXREG. While TXIF indicates the status of TXREG, TRMT (TXSTA<1>) 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. 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. The first data bit will be shifted out on the next available rising edge of the clock on the TX/CK pin. Data out is stable around the falling edge of the synchronous clock (Figure 14-9). The transmission can also be started by first loading TXREG and then setting TXEN. This is advantageous when slow baud rates are selected, since BRG is kept in RESET when the TXEN, CREN, and SREN bits are clear. Setting the TXEN bit will start the BRG, creating a shift clock immediately. Normally when transmission is first started, the TSR is empty, so a transfer to TXREG will result in an immediate transfer to the TSR, resulting in an empty TXREG. Back-to-back transfers are possible.

Clearing TXEN during a transmission will cause the transmission to be aborted and will reset the transmitter. The RX/DT and TX/CK pins will revert to hi-impedance. If either CREN or SREN are set during a transmission, the transmission is aborted and the RX/DT pin reverts to a hi-impedance state (for a reception). The TX/CK pin will remain an output if the CSRC bit is set (internal clock). The transmitter logic is not reset, although it is disconnected from the pins. In order to reset the transmitter, the user has to clear the TXEN bit. If the SREN bit is set (to interrupt an ongoing transmission and receive a single word), then after the single word is received, SREN will be cleared and the serial port will revert back to transmitting, since the TXEN bit is still set. The DT line will immediately switch from hi-impedance Receive mode to transmit and start driving. To avoid this, TXEN should be cleared.

In order to select 9-bit transmission, the TX9 (TXSTA<6>) bit should be set and the ninth bit should be written to TX9D (TXSTA<0>). The ninth bit must be written before writing the 8-bit data to 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). If the TSR was empty and TXREG was written before writing the “new” TX9D, the “present” value of TX9D is loaded.

Steps to follow when setting up a Synchronous Master Transmission:

1. Initialize the SPBRG register for the appropriate baud rate (see Baud Rate Generator Section for details).
2. Enable the synchronous master serial port by setting the SYNC, SPEN, and CSRC bits.
3. Ensure that the CREN and SREN bits are clear (these bits override transmission when set).
4. If interrupts are desired, then set the TXIE bit (the GLINTD bit must be clear and the PEIE bit must be set).
5. If 9-bit transmission is desired, then set the TX9 bit.
6. If 9-bit transmission is selected, the ninth bit should be loaded in TX9D.
7. Start transmission by loading data to the TXREG register.
8. Enable the transmission by setting TXEN.

Writing the transmit data to the TXREG, then enabling the transmit (setting TXEN), allows transmission to start sooner than doing these two events in the reverse order.

Note:	To terminate a transmission, either clear the SPEN bit, or the TXEN bit. This will reset the transmit logic, so that it will be in the proper state when transmit is re-enabled.

REGISTER 15-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS: 13h, BANK 6)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

bit 7 **SMP:** Sample bit

SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time

SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode

In I^2C Master or Slave mode:
1 = Slew rate control disabled for Standard Speed mode (100 kHz and 1 MHz)
0 = Slew rate control enabled for High Speed mode (400 kHz)

bit 6 **CKE**: SPI Clock Edge Select (Figure 15-6, Figure 15-8 and Figure 15-9)

CKP = 0:
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK

CKP = 1:
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK

bit 5 **D/$\overline{A}$**: Data/$\overline{Address}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P**: STOP bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a STOP bit has been detected last (this bit is '0' on RESET)
0 = STOP bit was not detected last

bit 3 **S**: START bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a START bit has been detected last (this bit is '0' on RESET)
0 = START bit was not detected last

bit 2 **R/$\overline{W}$**: Read/Write bit Information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next START bit, STOP bit, or not ACK bit.

In I^2C Slave mode:
1 = Read
0 = Write

In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress
Or'ing this bit with SEN, RSEN, PEN, RCEN, or ACKEN will indicate if the MSSP is in IDLE mode.

bit 1 **UA**: Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0 **BF**: Buffer Full Status bit

Receive (SPI and I^2C modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty

Transmit (I^2C mode only)
1 = Data transmit in progress (does not include the $\overline{ACK}$ and STOP bits), SSPBUF is full
0 = Data transmit complete (does not include the $\overline{ACK}$ and STOP bits), SSPBUF is empty

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

15.2.12 I^2C MASTER MODE RECEPTION

Master mode reception is enabled by programming the receive enable bit, RCEN (SSPCON2<3>).

Note:	The SSP Module must be in an IDLE STATE before the RCEN bit is set, or the RCEN bit will be disregarded.

The baud rate generator begins counting and on each rollover, the state of the SCL pin changes (high to low/ low to high) and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag is set, the SSPIF is set and the baud rate generator is suspended from counting, holding SCL low. The SSP is now in IDLE state, awaiting the next command. When the buffer is read by the CPU, the BF flag is automatically cleared. The user can then send an acknowledge bit at the end of reception, by setting the acknowledge sequence enable bit, ACKEN (SSPCON2<4>).

15.2.12.1 BF Status Flag

In receive operation, BF is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when SSPBUF is read.

15.2.12.2 SSPOV Status Flag

In receive operation, SSPOV is set when 8 bits are received into the SSPSR, and the BF flag is already set from a previous reception.

15.2.12.3 WCOL Status Flag

If the user writes the SSPBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 15-36: BUS COLLISION DURING START CONDITION (SCL = 0)

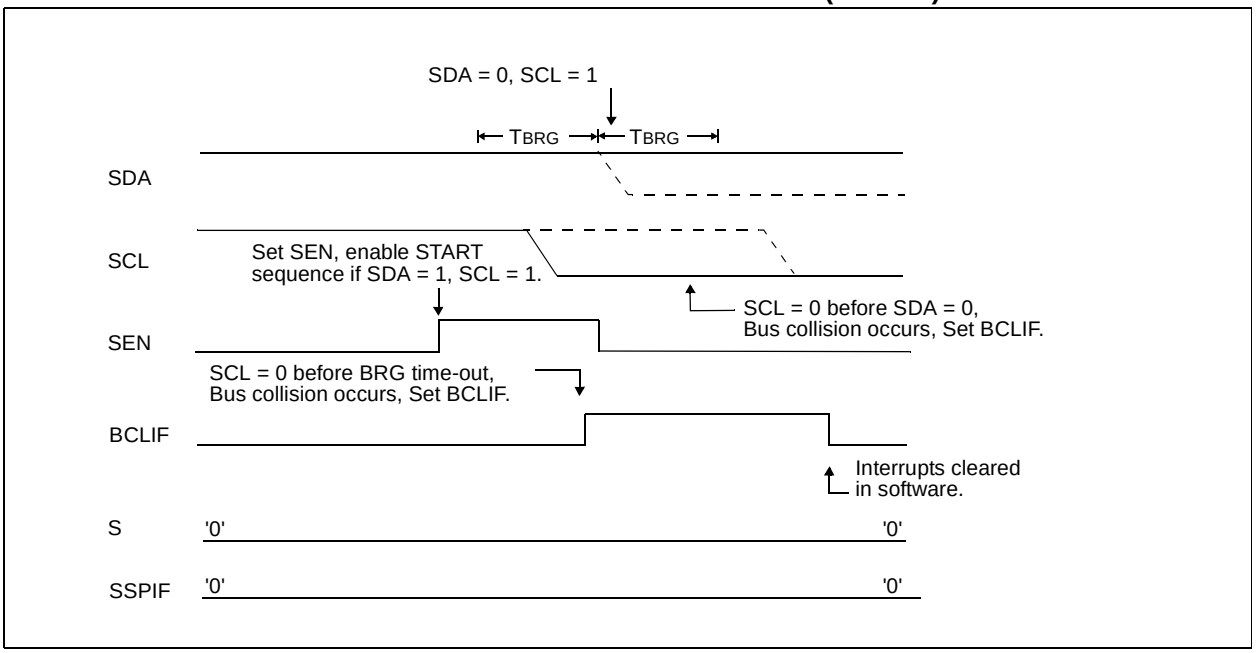
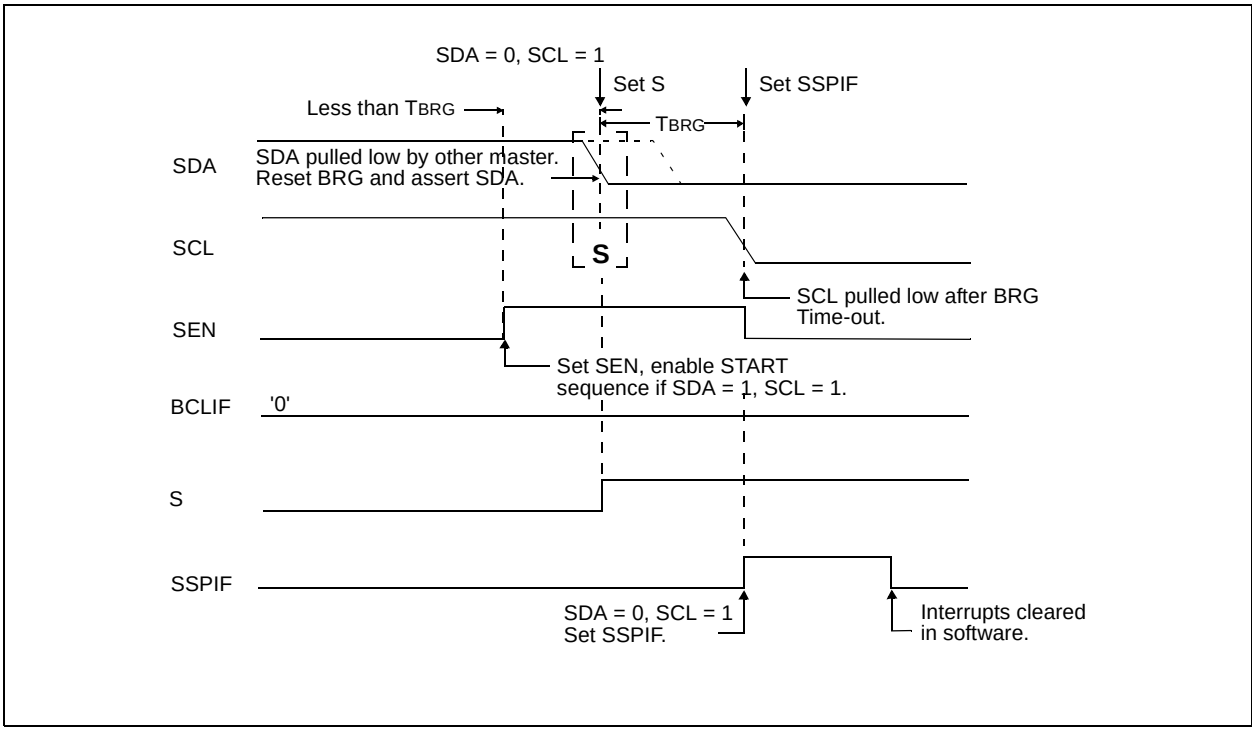


FIGURE 15-37: BRG RESET DUE TO SDA COLLISION DURING START CONDITION



17.0 SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC17CXXX family has a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator Selection (Section 4.0)
- RESET (Section 5.0)
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts (Section 6.0)
- Watchdog Timer (WDT)
- SLEEP mode
- Code protection

The PIC17CXXX has a Watchdog Timer which can be shut-off only through EPROM bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on POR and BOR. One is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 96 ms (nominal) on power-up only, designed to keep the part in RESET while the power supply stabilizes. With these two timers on-chip, most applications need no external RESET circuitry.

The SLEEP mode is designed to offer a very low current power-down mode. The user can wake from SLEEP through external RESET, Watchdog Timer Reset, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost, while the LF crystal option saves power. Configuration bits are used to select various options. This configuration word has the format shown in Figure 17-1.

REGISTER 17-1: CONFIGURATION WORDS

High (H) Table Read Addr.	U-x	R/P-1	R/P-1	U-x	U-x	U-x	U-x	U-x	U-x
FE0Fh - FE08h	—	PM2	BODEN	—	—	—	—	—	—
	bit 15 bit 8	bit 7							bit 0

Low (L) Table Read Addr.	U-x	U-x	R/P-1	U-x	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FE07h - FE00h	—	—	PM1	—	PM0	WDTPS1	WDTPS0	FOSC1	FOSC0
	bit 15 bit 8	bit 7							bit 0

bits 7H, 6L, 4L	**PM2, PM1, PM0:** Processor Mode Select bits 111 = Microprocessor mode 110 = Microcontroller mode 101 = Extended Microcontroller mode 000 = Code Protected Microcontroller mode
bit 6H	**BODEN:** Brown-out Detect Enable 1 = Brown-out Detect circuitry is enabled 0 = Brown-out Detect circuitry is disabled
bits 3L:2L	**WDTPS1:WDTPS0**: WDT Postscaler Select bits 11 = WDT enabled, postscaler = 1 10 = WDT enabled, postscaler = 256 01 = WDT enabled, postscaler = 64 00 = WDT disabled, 16-bit overflow timer
bits 1L:0L	**FOSC1:FOSC0**: Oscillator Select bits 11 = EC oscillator 10 = XT oscillator 01 = RC oscillator 00 = LF oscillator
Shaded bits (—)	**Reserved**

18.2 Q Cycle Activity

Each instruction cycle (TCY) is comprised of four Q cycles (Q1-Q4). The Q cycle is the same as the device oscillator cycle (TOSC). The Q cycles provide the timing/ designation for the Decode, Read, Process Data, Write, etc., of each instruction cycle. The following diagram shows the relationship of the Q cycles to the instruction cycle.

The four Q cycles that make up an instruction cycle (TCY) can be generalized as:

Q1: Instruction Decode Cycle or forced No operation

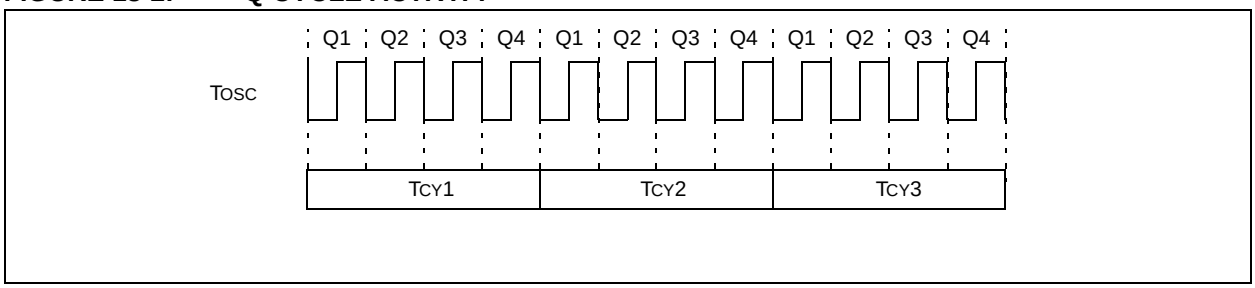
Q2: Instruction Read Cycle or No operation

Q3: Process the Data

Q4: Instruction Write Cycle or No operation

Each instruction will show the detailed Q cycle operation for the instruction.

FIGURE 18-2: Q CYCLE ACTIVITY



ADDLW **ADD Literal to WREG**

Syntax: [*label*] ADDLW k

Operands: $0 \le k \le 255$

Operation: (WREG) + k → (WREG)

Status Affected: OV, C, DC, Z

Encoding:

1011	0001	kkkk	kkkk

Description: The contents of WREG are added to the 8-bit literal 'k' and 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: `ADDLW 0x15`

Before Instruction
WREG = 0x10

After Instruction
WREG = 0x25

ADDWF **ADD WREG to f**

Syntax: [*label*] ADDWF f,d

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

Operation: (WREG) + (f) → (dest)

Status Affected: OV, C, DC, Z

Encoding:

0000	111d	ffff	ffff

Description: Add WREG to 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'	Process Data	Write to destination

Example: `ADDWF REG, 0`

Before Instruction
WREG = 0x17
REG = 0xC2

After Instruction
WREG = 0xD9
REG = 0xC2

MULLW **Multiply Literal with WREG**

Syntax: [*label*] MULLW k

Operands: $0 \le k \le 255$

Operation: (k x WREG) → PRODH:PRODL

Status Affected: None

Encoding:

1011	1100	kkkk	kkkk

Description: An unsigned multiplication is carried out between the contents of WREG and the 8-bit literal 'k'. The 16-bit result is placed in PRODH:PRODL register pair. PRODH contains the high byte.

WREG is unchanged.

None of the status flags are affected.

Note that neither overflow, nor carry is possible in this operation. A zero result is possible, but not detected.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write registers PRODH: PRODL

Example: `MULLW 0xC4`

Before Instruction
- WREG = 0xE2
- PRODH = ?
- PRODL = ?

After Instruction
- WREG = 0xC4
- PRODH = 0xAD
- PRODL = 0x08

MULWF **Multiply WREG with f**

Syntax: [*label*] MULWF f

Operands: $0 \le f \le 255$

Operation: (WREG x f) → PRODH:PRODL

Status Affected: None

Encoding:

0011	0100	ffff	ffff

Description: An unsigned multiplication is carried out between the contents of WREG and the register file location 'f'. The 16-bit result is stored in the PRODH:PRODL register pair. PRODH contains the high byte.

Both WREG and 'f' are unchanged.

None of the status flags are affected.

Note that neither overflow, nor carry is possible in this operation. A zero result is possible, but not detected.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write registers PRODH: PRODL

Example: `MULWF REG`

Before Instruction
- WREG = 0xC4
- REG = 0xB5
- PRODH = ?
- PRODL = ?

After Instruction
- WREG = 0xC4
- REG = 0xB5
- PRODH = 0x8A
- PRODL = 0x94

NEGW **Negate W**

Syntax: [*label*] NEGW f,s

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

Operation: $\overline{WREG} + 1 \rightarrow (f)$;
$\overline{WREG} + 1 \rightarrow s$

Status Affected: OV, C, DC, Z

Encoding:

0010	110s	ffff	ffff

Description: WREG is negated using two's complement. If 's' is 0, the result is placed in WREG and data memory location 'f'. If 's' is 1, the result is placed only in data memory location 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `NEGW REG,0`

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

After Instruction
WREG = 1100 0110 [0xC6]
REG = 1100 0110 [0xC6]

NOP **No Operation**

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Encoding:

0000	0000	0000	0000

Description: No operation.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation

Example:

None.

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

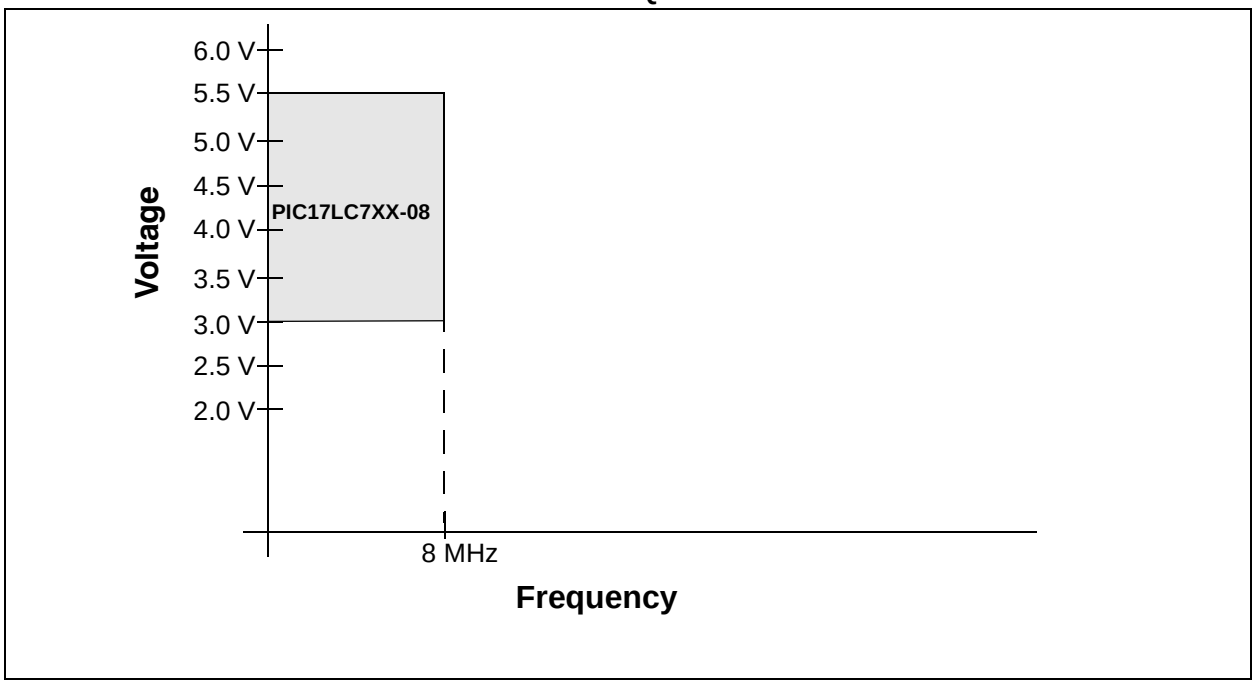
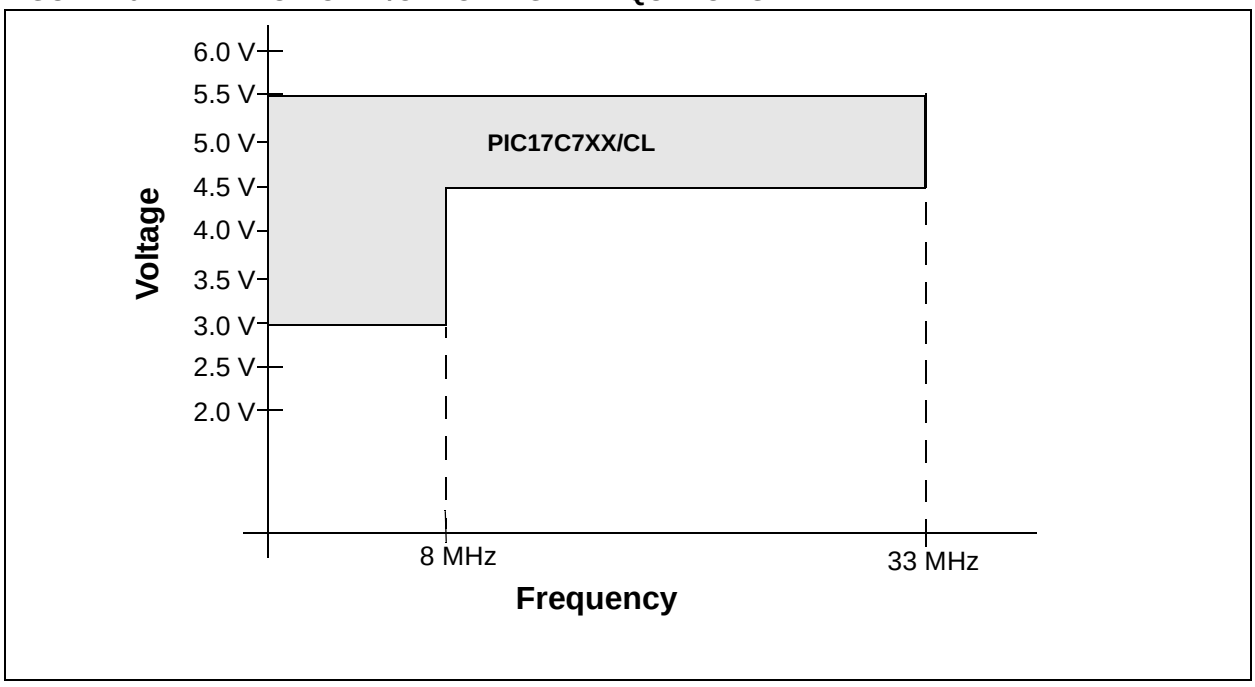


FIGURE 20-4: PIC17C7XX/CL VOLTAGE-FREQUENCY GRAPH



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.

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 Leakage Current (Notes 2, 3)					
D060	IIL	I/O ports (except RA2, RA3)	–	–	±1	µA	Vss ≤ VPIN ≤ VDD, I/O Pin (in digital mode) at hi-impedance PORTB weak pull-ups disabled
D061		$\overline{\text{MCLR}}$, TEST	–	–	±2	µA	VPIN = Vss or VPIN = VDD
D062		RA2, RA3			±2	µA	Vss ≤ VRA2, VRA3 ≤ 12V
D063		OSC1 (EC, RC modes)	–	–	±1	µA	Vss ≤ VPIN ≤ VDD
D063B		OSC1 (XT, LF modes)	–	–	VPIN	µA	RF ≥ 1 MΩ
D064		$\overline{\text{MCLR}}$, TEST	–	–	25	µA	VMCLR = VPP = 12V (when not programming)
D070	IPURB	**PORTB Weak Pull-up Current**	85	130	260	µA	VPIN = VSS, $\overline{\text{RBPU}}$ = 0 4.5V ≤ VDD ≤ 5.5V
		Output Low Voltage					
D080	VOL	I/O ports					IOL = VDD/1.250 mA
			–	–	0.1VDD	V	4.5V ≤ VDD ≤ 5.5V
			–	–	0.1VDD	V	VDD = 3.0V
D081		with TTL buffer	–	–	0.4	V	IOL = 6 mA, VDD = 4.5V **(Note 6)**
D082		RA2 and RA3	–	–	3.0	V	IOL = 60.0 mA, VDD = 5.5V
			–	–	0.6	V	IOL = 60.0 mA, VDD = 4.5V
D083		OSC2/CLKOUT	–	–	0.4	V	IOL = 1 mA, VDD = 4.5V
D084		(RC and EC osc modes)	–	–	0.1VDD	V	IOL = VDD/5 mA (PIC17LC7XX only)
		Output High Voltage (Note 3)					
D090	VOH	I/O ports (except RA2 and RA3)					IOH = -VDD/2.5 mA
			0.9VDD	–	–	V	4.5V ≤ VDD ≤ 5.5V
			0.9VDD	–	–	V	VDD = 3.0V
D091		with TTL buffer	2.4	–	–	V	IOH = -6.0 mA, VDD = 4.5V **(Note 6)**
D093		OSC2/CLKOUT	2.4	–	–	V	IOH = -5 mA, VDD = 4.5V
D094		(RC and EC osc modes)	0.9VDD	–	–	V	IOH = -VDD/5 mA (PIC17LC7XX only)

† 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 $\overline{\text{MCLR}}$ 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 $\overline{\text{MCLR}}$/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.

Param No.	Sym	Characteristic		Min	Max	Units	Conditions
110	Tbuf	Bus free time	100 kHz mode	4.7	—	ms	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	ms	
			1 MHz mode(1)	0.5	—	ms	
D102	Cb	Bus capacitive loading		—	400	pF	

Note 1: Maximum pin capacitance = 10 pF for all I^2C pins.

2: A fast mode (400 KHz) I^2C bus device can be used in a standard mode I^2C 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 20-19: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

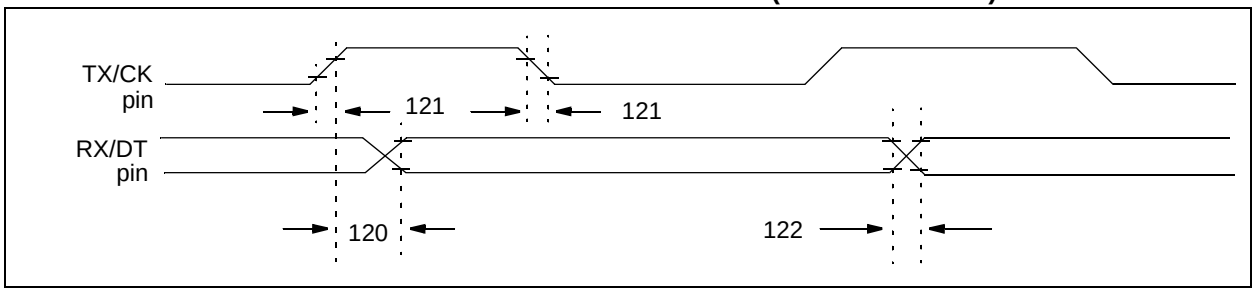


TABLE 20-14: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE) Clock high to data out valid	PIC17**C**XXX	—	—	50	ns	
			PIC17**LC**XXX	—	—	75	ns	
121	TckRF	Clock out rise time and fall time (Master mode)	PIC17**C**XXX	—	—	25	ns	
			PIC17**LC**XXX	—	—	40	ns	
122	TdtRF	Data out rise time and fall time	PIC17**C**XXX	—	—	25	ns	
			PIC17**LC**XXX	—	—	40	ns	

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

FIGURE 20-24: MEMORY INTERFACE WRITE TIMING

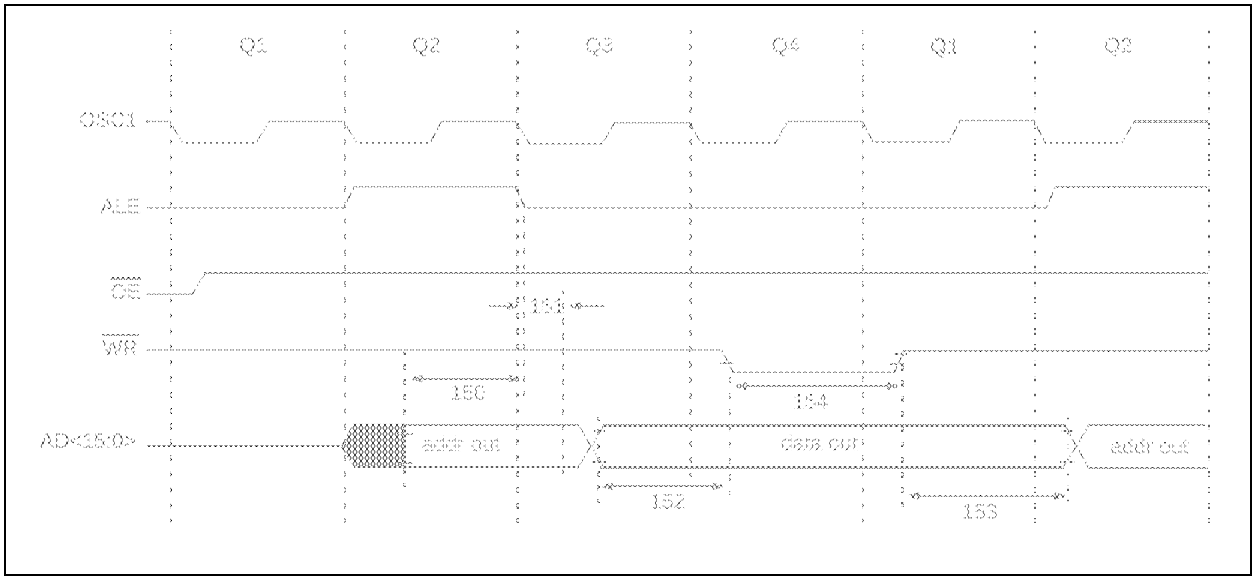
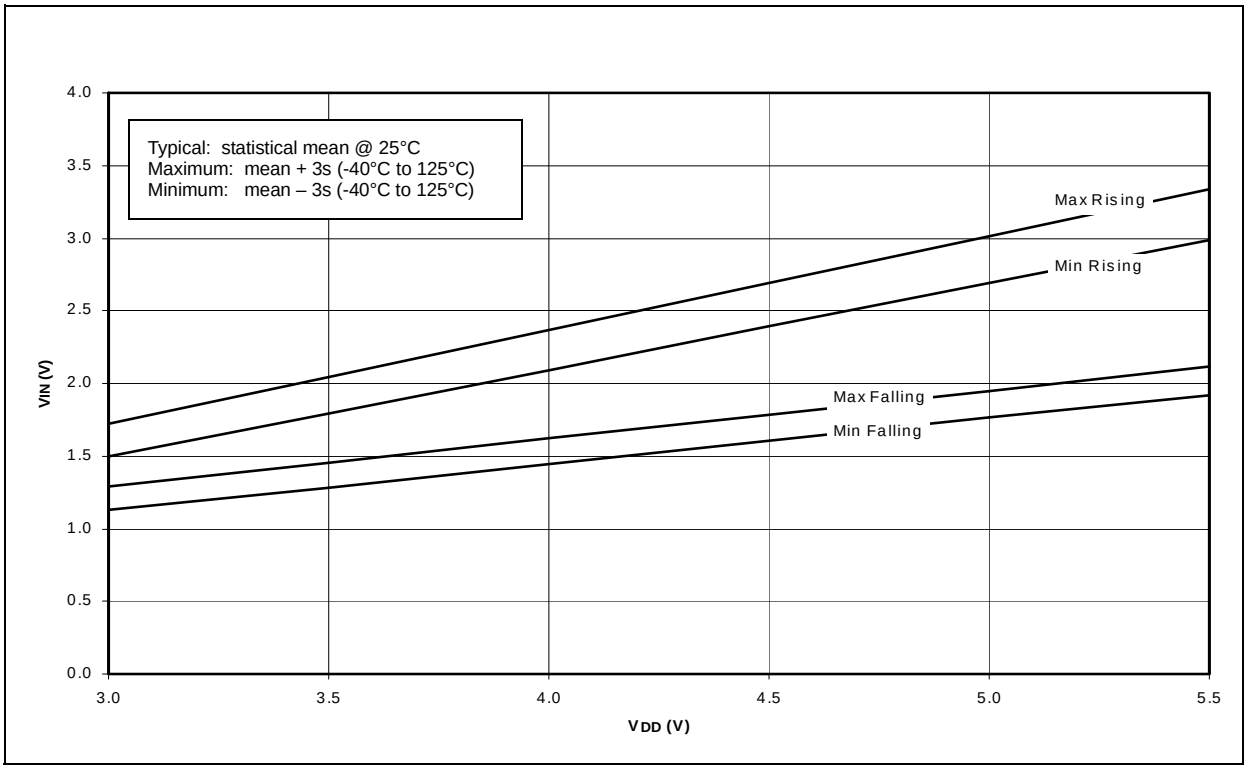


TABLE 20-20: MEMORY INTERFACE WRITE REQUIREMENTS

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 21-23: MAXIMUM AND MINIMUM VIN vs. VDD (I^2C Input, -40°C to +125°C)



READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager Total Pages Sent

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: **PIC17C7XX** Literature Number: **DS30289C**

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this data sheet easy to follow? If not, why?

4. What additions to the data sheet do you think would enhance the structure and subject?

5. What deletions from the data sheet could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?