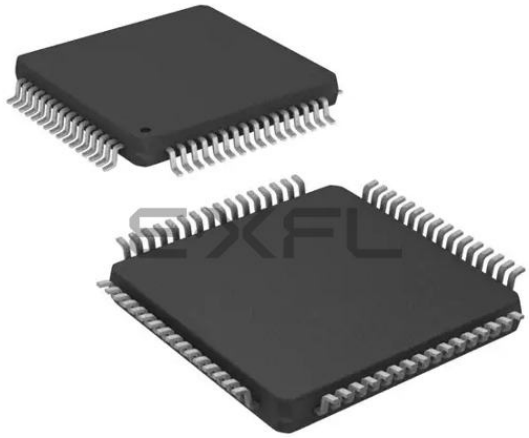


Welcome to E-XFL.COM

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	33MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	50
Program Memory Size	16KB (8K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	454 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c752-33-pt

FIGURE 3-1: PIC17C752/756A BLOCK DIAGRAM

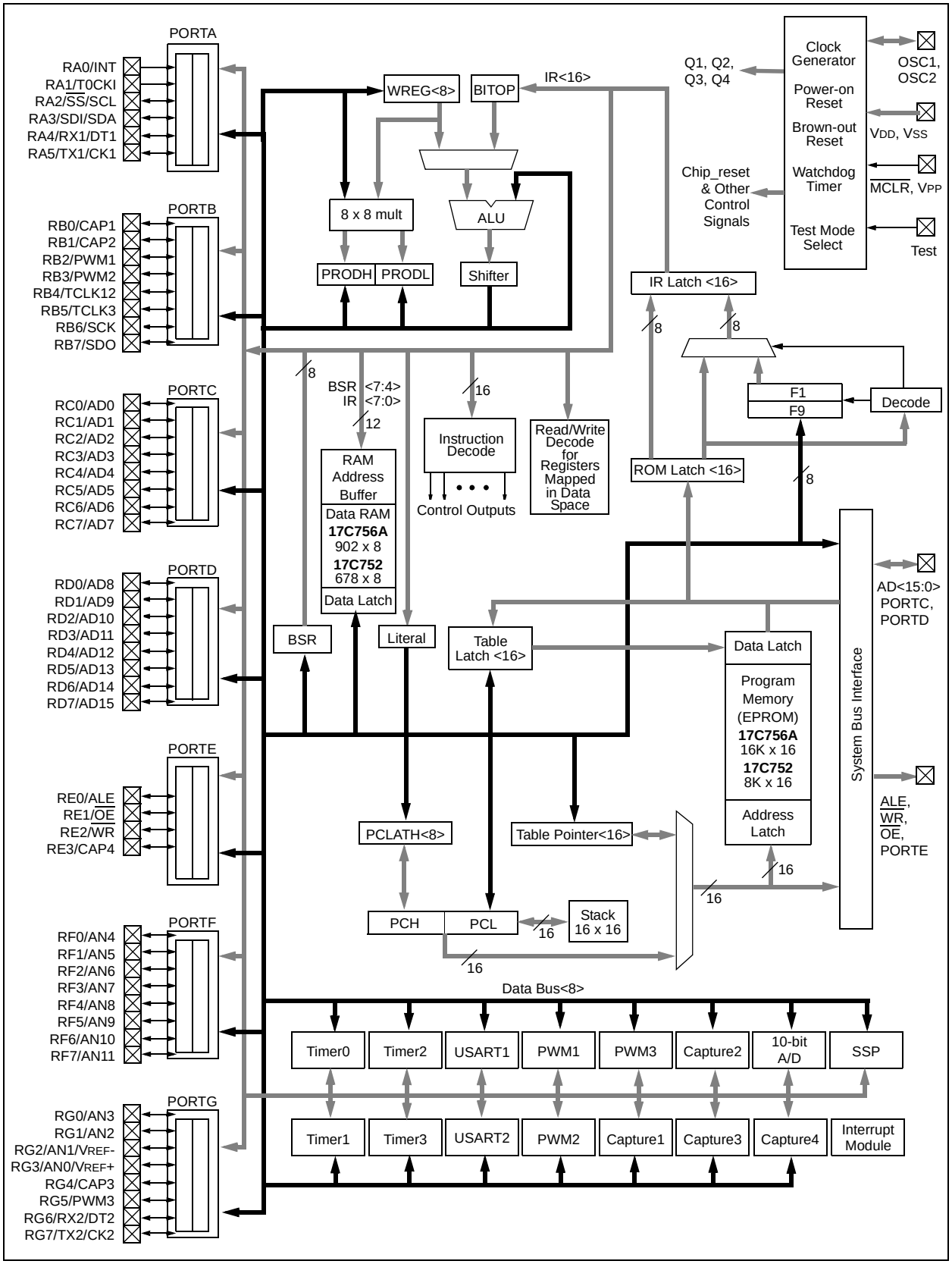


TABLE 7-3: SPECIAL FUNCTION REGISTERS (CONTINUED)

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
Bank 6											
10h	SSPADD	SSP Address Register in I^2C Slave mode. SSP Baud Rate Reload Register in I^2C Master mode								0000 0000	0000 0000
11h	SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
12h	SSPCON2	GCEN	AKSTAT	AKDT	AKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
13h	SSPSTAT	SMP	CKE	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF	0000 0000	0000 0000
14h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
15h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
16h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
17h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Bank 7											
10h	PW3DCL	DC1	DC0	TM2PW3	—	—	—	—	—	xx0- ----	uu0- ----
11h	PW3DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
12h	CA3L	Capture3 Low Byte								xxxx xxxx	uuuu uuuu
13h	CA3H	Capture3 High Byte								xxxx xxxx	uuuu uuuu
14h	CA4L	Capture4 Low Byte								xxxx xxxx	uuuu uuuu
15h	CA4H	Capture4 High Byte								xxxx xxxx	uuuu uuuu
16h	TCON3	—	CA4OVF	CA3OVF	CA4ED1	CA4ED0	CA3ED1	CA3ED0	PWM3ON	-000 0000	-000 0000
17h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Bank 8(3)											
10h(3)	DDRH	Data Direction Register for PORTH								1111 1111	1111 1111
11h(3)	PORTH(4)	RH7/ AN15	RH6/ AN14	RH5/ AN13	RH4/ AN12	RH3	RH2	RH1	RH0	xxxx xxxx	uuuu uuuu
12h(3)	DDRJ	Data Direction Register for PORTJ								1111 1111	1111 1111
13h(3)	PORTJ(4)	RJ7	RJ6	RJ5	RJ4	RJ3	RJ2	RJ1	RJ0	xxxx xxxx	uuuu uuuu
14h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
15h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
16h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
17h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Unbanked											
18h	PRODL	Low Byte of 16-bit Product (8 x 8 Hardware Multiply)								xxxx xxxx	uuuu uuuu
19h	PRODH	High Byte of 16-bit Product (8 x 8 Hardware Multiply)								xxxx xxxx	uuuu uuuu

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<15:8> whose contents are updated from, or transferred to, the upper byte of the program counter.

2: The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ status bits in CPUSTA are not affected by a $\overline{\text{MCLR}}$ Reset.

3: Bank 8 and associated registers are only implemented on the PIC17C76X devices.

4: This is the value that will be in the port output latch.

5: When the device is configured for Microprocessor or Extended Microcontroller mode, the operation of this port does not rely on these registers.

6: On any device RESET, these pins are configured as inputs.

7.7 Program Counter Module

The Program Counter (PC) is a 16-bit register. PCL, the low byte of the PC, is mapped in the data memory. PCL is readable and writable just as is any other register. PCH is the high byte of the PC and is not directly addressable. Since PCH is not mapped in data or program memory, an 8-bit register PCLATH (PC high latch) is used as a holding latch for the high byte of the PC. PCLATH is mapped into data memory. The user can read or write PCH through PCLATH.

The 16-bit wide PC is incremented after each instruction fetch during Q1 unless:

- Modified by a GOTO, CALL, LCALL, RETURN, RETLW, or RETFIE instruction
- Modified by an interrupt response
- Due to destination write to PCL by an instruction

"Skips" are equivalent to a forced NOP cycle at the skipped address.

Figure 7-7 and Figure 7-8 show the operation of the program counter for various situations.

FIGURE 7-7: PROGRAM COUNTER OPERATION

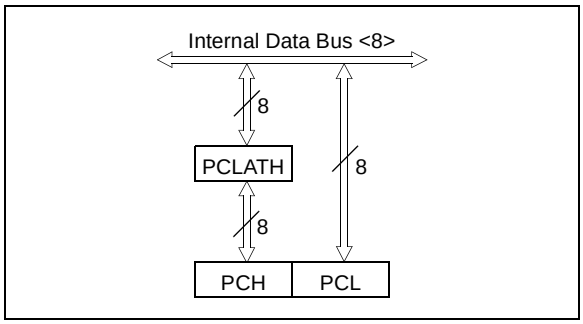
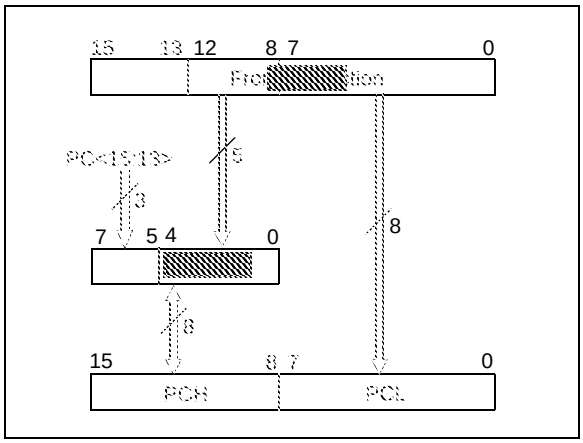


FIGURE 7-8: PROGRAM COUNTER USING THE CALL AND GOTO INSTRUCTIONS



Using Figure 7-7, the operations of the PC and PCLATH for different instructions are as follows:

a) LCALL instructions:
   An 8-bit destination address is provided in the instruction (opcode). PCLATH is unchanged.
   PCLATH → PCH
   Opcode<7:0> → PCL

b) Read instructions on PCL:
   Any instruction that reads PCL.
   PCL → data bus → ALU or destination
   PCH → PCLATH

c) Write instructions on PCL:
   Any instruction that writes to PCL.
   8-bit data → data bus → PCL
   PCLATH → PCH

d) Read-Modify-Write instructions on PCL:
   Any instruction that does a read-write-modify operation on PCL, such as ADDWF PCL.
   Read: PCL → data bus → ALU
   Write: 8-bit result → data bus → PCL
   PCLATH → PCH

e) RETURN instruction:
   Stack<MRU> → PC<15:0>

Using Figure 7-8, the operation of the PC and PCLATH for GOTO and CALL instructions is as follows:

CALL, GOTO instructions:
A 13-bit destination address is provided in the instruction (opcode).
Opcode<12:0> → PC<12:0>
PC<15:13> → PCLATH<7:5>
Opcode<12:8> → PCLATH<4:0>

The read-modify-write only affects the PCL with the result. PCH is loaded with the value in the PCLATH. For example, ADDWF PCL will result in a jump within the current page. If PC = 03F0h, WREG = 30h and PCLATH = 03h before instruction, PC = 0320h after the instruction. To accomplish a true 16-bit computed jump, the user needs to compute the 16-bit destination address, write the high byte to PCLATH and then write the low value to PCL.

The following PC related operations do not change PCLATH:

a) LCALL, RETLW, and RETFIE instructions.
b) Interrupt vector is forced onto the PC.
c) Read-modify-write instructions on PCL (e.g. BSF PCL).

TABLE 10-5: 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 10-6: 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 POR, BOR	$\overline{\text{MCLR}}$, WDT
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

10.7 PORTG and DDRG Registers

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

The lower four bits of PORTG are multiplexed with four channels of the 10-bit A/D converter.

The remaining bits of PORTG are multiplexed with peripheral output and inputs. RG4 is multiplexed with the CAP3 input, RG5 is multiplexed with the PWM3 output, RG6 and RG7 are multiplexed with the USART2 functions.

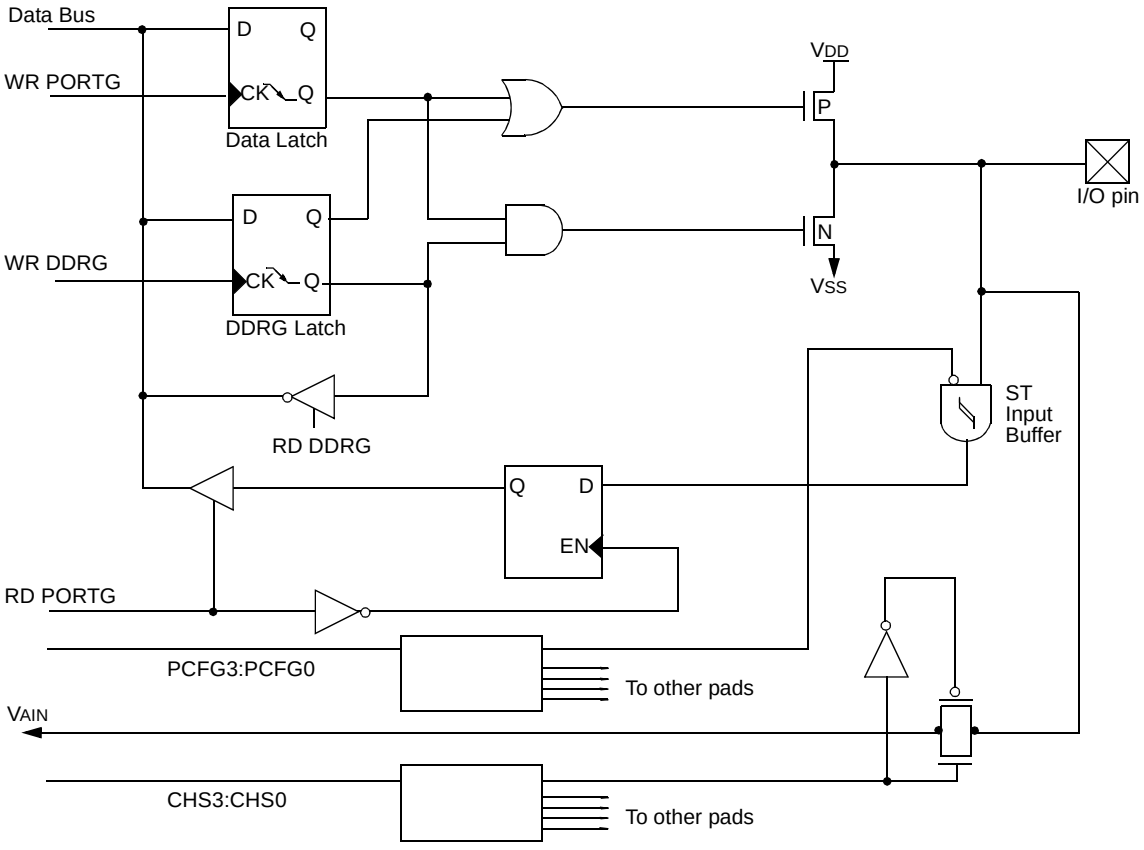
Upon RESET, RG3:RG0 is automatically configured as analog inputs and must be configured in software to be a digital I/O.

Example 10-7 shows the instruction sequence to initialize PORTG. 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 10-7: INITIALIZING PORTG

```
MOVLB  5              ; Select Bank 5
MOVLW  0x0E           ; Configure PORTG as
MOVPF  WREG, ADCON1   ; digital
CLRF   PORTG, F      ; Initialize PORTG data
                      ;   latches before
                      ;   the data direction
                      ;   register
MOVLW   0x03          ; Value used to init
                      ;   data direction
MOVWF   DDRG         ; Set RG<1:0> as inputs
                      ;   RG<7:2> as outputs
```

FIGURE 10-14: BLOCK DIAGRAM OF RG3:RG0



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

TABLE 10-13: PORTG FUNCTIONS

Name	Bit	Buffer Type	Function
RG0/AN3	bit0	ST	Input/output or analog input 3.
RG1/AN2	bit1	ST	Input/output or analog input 2.
RG2/AN1/V_{REF-}	bit2	ST	Input/output or analog input 1 or the ground reference voltage.
RG3/AN0/V_{REF+}	bit3	ST	Input/output or analog input 0 or the positive reference voltage.
RG4/CAP3	bit4	ST	Input/output or the Capture3 input pin.
RG5/PWM3	bit5	ST	Input/output or the PWM3 output pin.
RG6/RX2/DT2	bit6	ST	Input/output or the USART2 (SCI) Asynchronous Receive or USART2 (SCI) Synchronous Data.
RG7/TX2/CK2	bit7	ST	Input/output or the USART2 (SCI) Asynchronous Transmit or USART2 (SCI) Synchronous Clock.

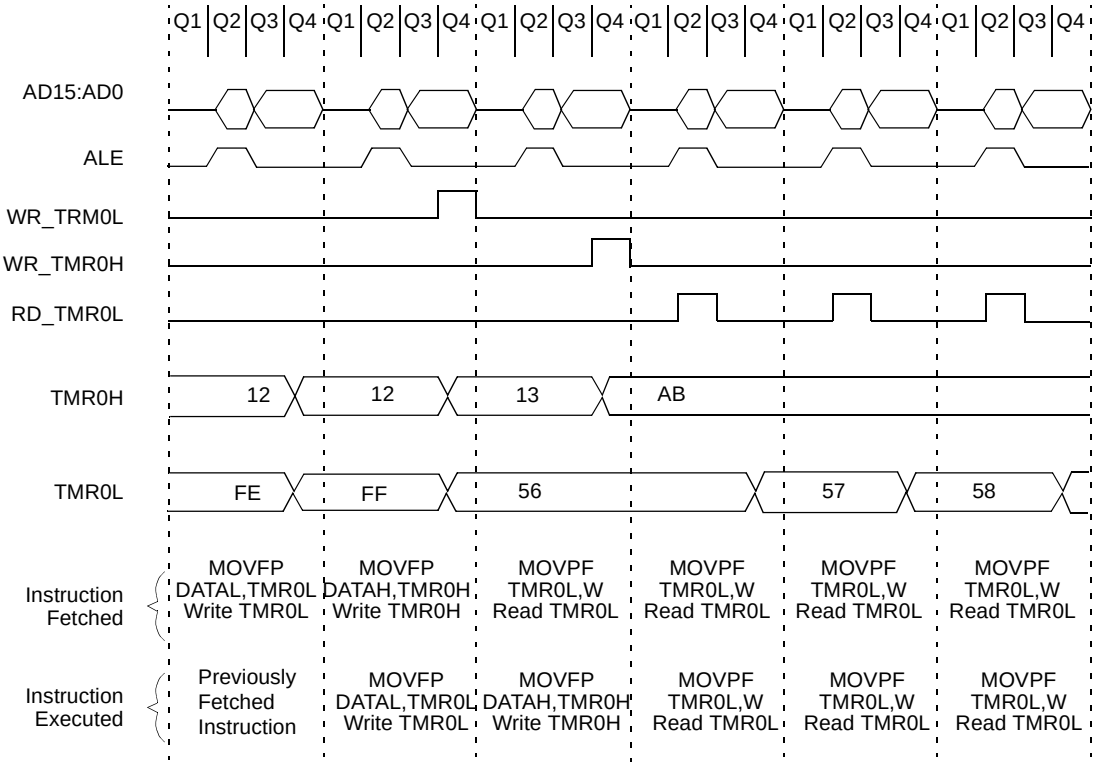
Legend: ST = Schmitt Trigger input

TABLE 10-14: REGISTERS/BITS ASSOCIATED WITH PORTG

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{MCLR}$, WDT
12h, Bank 5	DDRG	Data Direction Register for PORTG								1111 1111	1111 1111
13h, Bank 5	PORTG	RG7/ TX2/CK2	RG6/ RX2/DT2	RG5/ PWM3	RG4/ CAP3	RG3/ AN0	RG2/ AN1	RG1/ AN2	RG0/ AN3	xxxx 0000	uuuu 0000
15h, Bank 5	ADCON1	ADCS1	ADCS0	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0	000- 0000	000- 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by PORTG.

FIGURE 12-4: TMR0 READ/WRITE IN TIMER MODE



Note: In this example, old TMR0 value is 12FEh, new value of AB56h is written.

TABLE 12-1: REGISTERS/BITS ASSOCIATED WITH TIMER0

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
05h, Unbanked	T0STA	INTEDG	T0SE	T0CS	T0PS3	T0PS2	T0PS1	T0PS0	—	0000 000-	0000 000-
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	$\overline{\text{TO}}$	$\overline{\text{PD}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	--11 11qq	--11 qquu
07h, Unbanked	INTSTA	PEIF	T0CKIF	T0IF	INTF	PEIE	T0CKIE	T0IE	INTE	0000 0000	0000 0000
0Bh, Unbanked	TMR0L	TMR0 Register; Low Byte								xxxx xxxx	uuuu uuuu
0Ch, Unbanked	TMR0H	TMR0 Register; High Byte								xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as a '0', q = value depends on condition. Shaded cells are not used by Timer0.

13.0 TIMER1, TIMER2, TIMER3, PWMS AND CAPTURES

The PIC17C7XX has a wealth of timers and time based functions to ease the implementation of control applications. These time base functions include three PWM outputs and four Capture inputs.

Timer1 and Timer2 are two 8-bit incrementing timers, each with an 8-bit period register (PR1 and PR2, respectively) and separate overflow interrupt flags. Timer1 and Timer2 can operate either as timers (increment on internal FOSC/4 clock), or as counters (increment on falling edge of external clock on pin RB4/TCLK12). They are also software configurable to operate as a single 16-bit timer/counter. These timers are also used as the time base for the PWM (Pulse Width Modulation) modules.

Timer3 is a 16-bit timer/counter which uses the TMR3H and TMR3L registers. Timer3 also has two additional registers (PR3H/CA1H:PR3L/CA1L) that are configurable as a 16-bit period register or a 16-bit capture register. TMR3 can be software configured to increment from the internal system clock (FOSC/4), or from an external signal on the RB5/TCLK3 pin. Timer3 is the time base for all of the 16-bit captures.

Six other registers comprise the Capture2, Capture3, and Capture4 registers (CA2H:CA2L, CA3H:CA3L, and CA4H:CA4L).

Figure 13-1, Figure 13-2 and Figure 13-3 are the control registers for the operation of Timer1, Timer2 and Timer3, as well as PWM1, PWM2, PWM3, Capture1, Capture2, Capture3 and Capture4.

Table 13-1 shows the Timer resource requirements for these time base functions. Each timer is an open resource so that multiple functions may operate with it.

TABLE 13-1: TIME-BASE FUNCTION/ RESOURCE REQUIREMENTS

Time Base Function	Timer Resource
PWM1	Timer1
PWM2	Timer1 or Timer2
PWM3	Timer1 or Timer2
Capture1	Timer3
Capture2	Timer3
Capture3	Timer3
Capture4	Timer3

REGISTER 13-1: TCON1 REGISTER (ADDRESS: 16h, BANK 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CA2ED1	CA2ED0	CA1ED1	CA1ED0	T16	TMR3CS	TMR2CS	TMR1CS
bit 7							bit 0

bit 7-6 **CA2ED1:CA2ED0**: Capture2 Mode Select bits
00 = Capture on every falling edge
01 = Capture on every rising edge
10 = Capture on every 4th rising edge
11 = Capture on every 16th rising edge

bit 5-4 **CA1ED1:CA1ED0**: Capture1 Mode Select bits
00 = Capture on every falling edge
01 = Capture on every rising edge
10 = Capture on every 4th rising edge
11 = Capture on every 16th rising edge

bit 3 **T16**: Timer2:Timer1 Mode Select bit
1 = Timer2 and Timer1 form a 16-bit timer
0 = Timer2 and Timer1 are two 8-bit timers

bit 2 **TMR3CS**: Timer3 Clock Source Select bit
1 = TMR3 increments off the falling edge of the RB5/TCLK3 pin
0 = TMR3 increments off the internal clock

bit 1 **TMR2CS**: Timer2 Clock Source Select bit
1 = TMR2 increments off the falling edge of the RB4/TCLK12 pin
0 = TMR2 increments off the internal clock

bit 0 **TMR1CS**: Timer1 Clock Source Select bit
1 = TMR1 increments off the falling edge of the RB4/TCLK12 pin
0 = TMR1 increments off the internal clock

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

REGISTER 13-2: TCON2 REGISTER (ADDRESS: 17h, BANK 3)

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CA2OVF	CA1OVF	PWM2ON	PWM1ON	CA1/$\overline{PR3}$	TMR3ON	TMR2ON	TMR1ON
bit 7							bit 0

bit 7 **CA2OVF**: Capture2 Overflow Status bit
This bit indicates that the capture value had not been read from the capture register pair (CA2H:CA2L) before the next capture event occurred. The capture register retains the oldest unread capture value (last capture before overflow). Subsequent capture events will not update the capture register with the TMR3 value until the capture register has been read (both bytes).
1 = Overflow occurred on Capture2 register
0 = No overflow occurred on Capture2 register

bit 6 **CA1OVF**: Capture1 Overflow Status bit
This bit indicates that the capture value had not been read from the capture register pair (PR3H/CA1H:PR3L/CA1L), before the next capture event occurred. The capture register retains the oldest unread capture value (last capture before overflow). Subsequent capture events will not update the capture register with the TMR3 value until the capture register has been read (both bytes).
1 = Overflow occurred on Capture1 register
0 = No overflow occurred on Capture1 register

bit 5 **PWM2ON**: PWM2 On bit
1 = PWM2 is enabled
(The RB3/PWM2 pin ignores the state of the DDRB<3> bit.)
0 = PWM2 is disabled
(The RB3/PWM2 pin uses the state of the DDRB<3> bit for data direction.)

bit 4 **PWM1ON**: PWM1 On bit
1 = PWM1 is enabled
(The RB2/PWM1 pin ignores the state of the DDRB<2> bit.)
0 = PWM1 is disabled
(The RB2/PWM1 pin uses the state of the DDRB<2> bit for data direction.)

bit 3 **CA1/$\overline{PR3}$**: CA1/$\overline{PR3}$ Register Mode Select bit
1 = Enables Capture1
(PR3H/CA1H:PR3L/CA1L is the Capture1 register. Timer3 runs without a period register.)
0 = Enables the Period register
(PR3H/CA1H:PR3L/CA1L is the Period register for Timer3.)

bit 2 **TMR3ON**: Timer3 On bit
1 = Starts Timer3
0 = Stops Timer3

bit 1 **TMR2ON**: Timer2 On bit
This bit controls the incrementing of the TMR2 register. When TMR2:TMR1 form the 16-bit timer (T16 is set), TMR2ON must be set. This allows the MSB of the timer to increment.
1 = Starts Timer2 (must be enabled if the T16 bit (TCON1<3>) is set)
0 = Stops Timer2

bit 0 **TMR1ON**: Timer1 On bit

When T16 is set (in 16-bit Timer mode):
1 = Starts 16-bit TMR2:TMR1
0 = Stops 16-bit TMR2:TMR1

When T16 is clear (in 8-bit Timer mode:
1 = Starts 8-bit Timer1
0 = Stops 8-bit Timer1

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

13.1.3 USING PULSE WIDTH MODULATION (PWM) OUTPUTS WITH TIMER1 AND TIMER2

Three high speed pulse width modulation (PWM) outputs are provided. The PWM1 output uses Timer1 as its time base, while PWM2 and PWM3 may independently be software configured to use either Timer1 or Timer2 as the time base. The PWM outputs are on the RB2/PWM1, RB3/PWM2 and RG5/PWM3 pins.

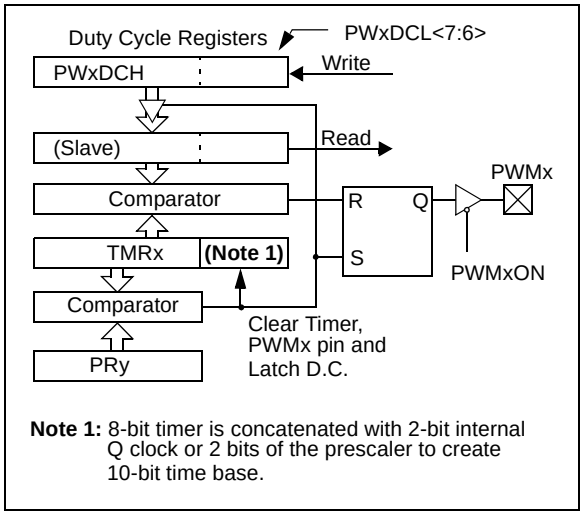
Each PWM output has a maximum resolution of 10-bits. At 10-bit resolution, the PWM output frequency is 32.2 kHz (@ 32 MHz clock) and at 8-bit resolution the PWM output frequency is 128.9 kHz. The duty cycle of the output can vary from 0% to 100%.

Figure 13-3 shows a simplified block diagram of a PWM module.

The duty cycle registers are double buffered for glitch free operation. Figure 13-4 shows how a glitch could occur if the duty cycle registers were not double buffered.

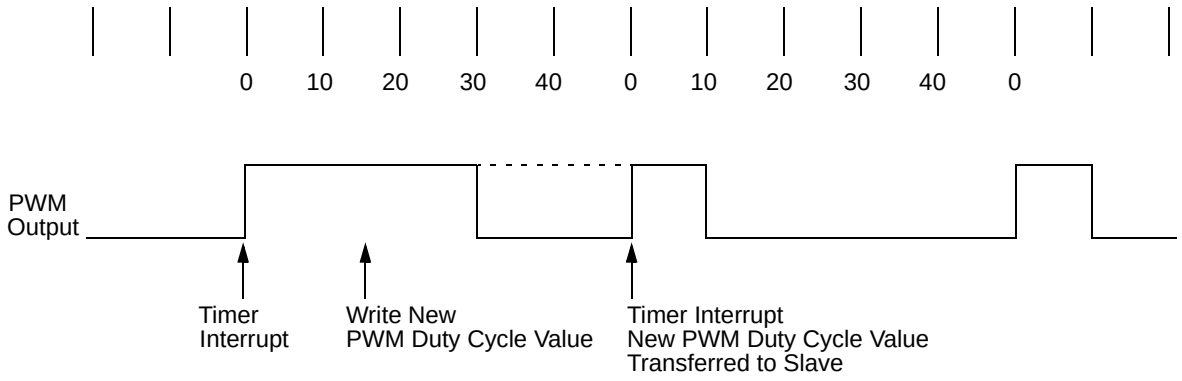
The user needs to set the PWM1ON bit (TCON2<4>) to enable the PWM1 output. When the PWM1ON bit is set, the RB2/PWM1 pin is configured as PWM1 output and forced as an output, irrespective of the data direction bit (DDRB<2>). When the PWM1ON bit is clear, the pin behaves as a port pin and its direction is controlled by its data direction bit (DDRB<2>). Similarly, the PWM2ON (TCON2<5>) bit controls the configuration of the RB3/PWM2 pin and the PWM3ON (TCON3<0>) bit controls the configuration of the RG5/PWM3 pin.

FIGURE 13-3: SIMPLIFIED PWM BLOCK DIAGRAM



Note 1: 8-bit timer is concatenated with 2-bit internal Q clock or 2 bits of the prescaler to create 10-bit time base.

FIGURE 13-4: PWM OUTPUT (NOT BUFFERED)



Note: The dotted line shows PWM output if duty cycle registers were not double buffered. If the new duty cycle is written after the timer has passed that value, then the PWM does not reset at all during the current cycle, causing a "glitch".

In this example, PWM period = 50. Old duty cycle is 30. New duty cycle value is 10.

15.2.1.3 Slave Transmission

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit, and the SCL pin is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then SCL pin should be enabled by setting bit CKP (SSPCON1<4>). The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 15-13).

An SSP interrupt is generated for each data transfer byte. The SSPIF flag bit must be cleared in software, and the SSPSTAT register is used to determine the status of the byte transfer. The SSPIF flag bit is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line was high (not $\overline{ACK}$), then the data transfer is complete. When the not $\overline{ACK}$ is latched by the slave, the slave logic is reset and the slave then monitors for another occurrence of the START bit. If the SDA line was low ($\overline{ACK}$), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then, the SCL pin should be enabled by setting the CKP bit.

FIGURE 15-12: I²C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)

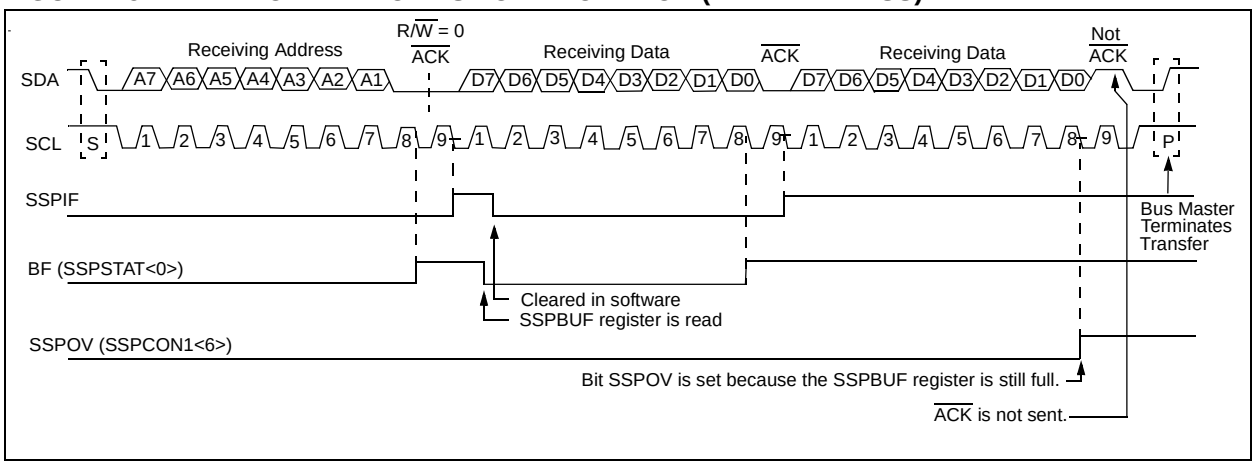


FIGURE 15-13: I²C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)

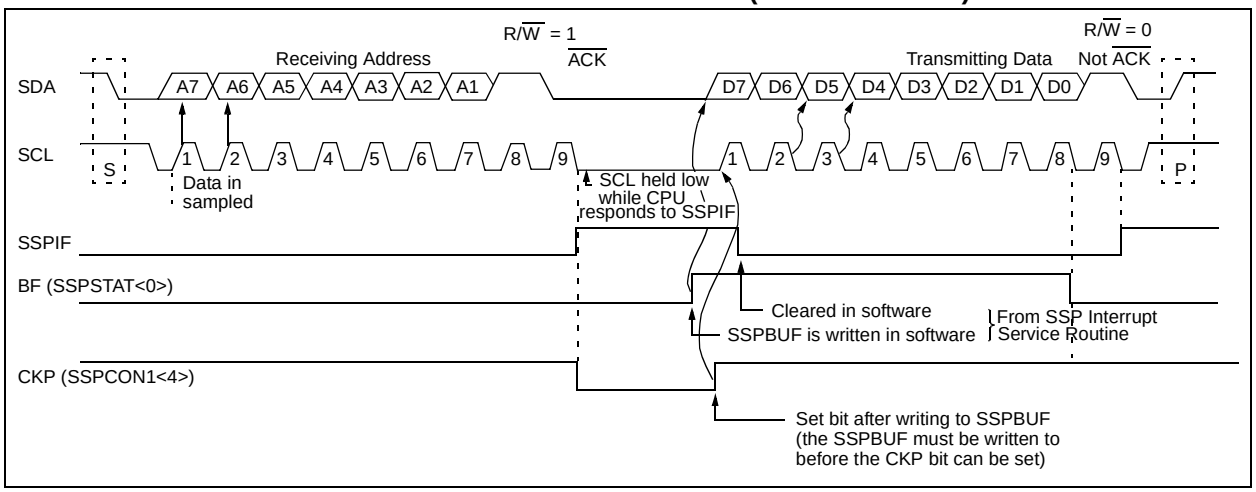
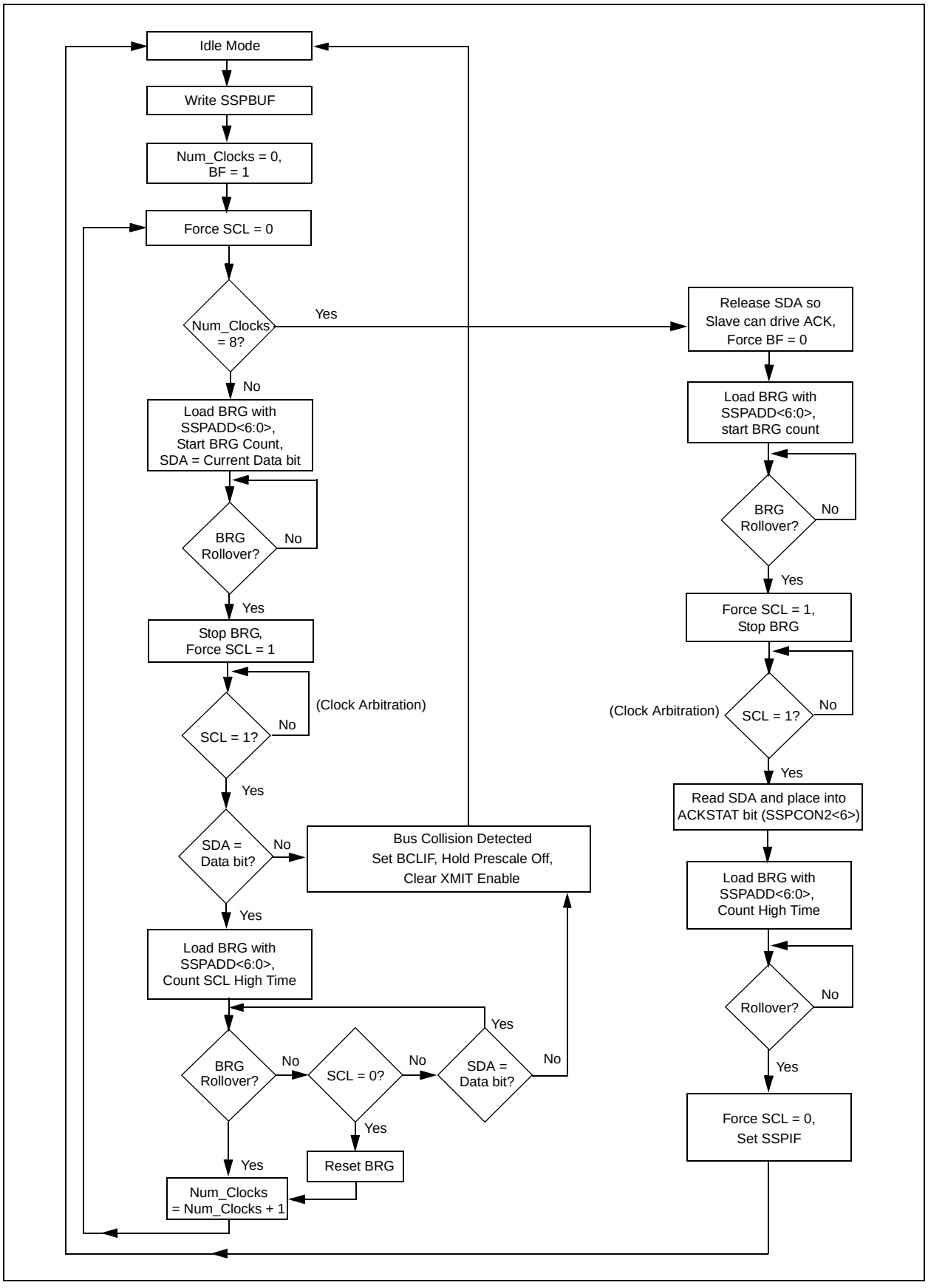


FIGURE 15-25: MASTER TRANSMIT FLOW CHART



REGISTER 16-2: ADCON1 REGISTER (ADDRESS 15h, BANK 5)

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS1	ADCS0	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

bit 7-6 **ADCS1:ADCS0**: A/D Conversion Clock Select bits
00 = FOSC/8
01 = FOSC/32
10 = FOSC/64
11 = FRC (clock derived from an internal RC oscillator)

bit 5 **ADFM**: A/D Result Format Select
1 = Right justified. 6 Most Significant bits of ADRESH are read as '0'.
0 = Left justified. 6 Least Significant bits of ADRESL are read as '0'.

bit 4 **Unimplemented**: Read as '0'

bit 3-1 **PCFG3:PCFG1**: A/D Port Configuration Control bits

bit 0 **PCFG0**: A/D Voltage Reference Select bit
1 = A/D reference is the VREF+ and VREF- pins
0 = A/D reference is AVDD and AVSS

Note: When this bit is set, ensure that the A/D voltage reference specifications are met.

PCFG3:PCFG0	AN15	AN14	AN13	AN12	AN11	AN10	AN9	AN8	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0
000x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
001x	D	A	A	A	A	A	A	A	D	A	A	A	A	A	A	A
010x	D	D	A	A	A	A	A	A	D	D	A	A	A	A	A	A
011x	D	D	D	A	A	A	A	A	D	D	D	A	A	A	A	A
100x	D	D	D	D	A	A	A	A	D	D	D	D	A	A	A	A
101x	D	D	D	D	D	A	A	A	D	D	D	D	D	A	A	A
110x	D	D	D	D	D	D	A	A	D	D	D	D	D	D	A	A
111x	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

A = Analog input D = Digital I/O

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

DCFSNZ **Decrement f, skip if not 0**

Syntax: [*label*] DCFSNZ f,d

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

Operation: (f) – 1 → (dest);
skip if not 0

Status Affected: None

Encoding:

0010	011d	ffff	ffff

Description: The contents of register 'f' are decremented. If 'd' is 0, the result is placed in WREG. If 'd' is 1, the result is placed back in register 'f'.
If the result is not 0, the next instruction, which is already fetched is discarded and a NOP is executed instead, making it a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

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

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

Example:
```
HERE    DCFSNZ  TEMP, 1
ZERO    :
NZERO   :
```

Before Instruction
TEMP_VALUE = ?

After Instruction
TEMP_VALUE = TEMP_VALUE - 1,
If TEMP_VALUE = 0;
PC = Address (ZERO)
If TEMP_VALUE ≠ 0;
PC = Address (NZERO)

GOTO **Unconditional Branch**

Syntax: [*label*] GOTO k

Operands: $0 \le k \le 8191$

Operation: k → PC<12:0>;
k<12:8> → PCLATH<4:0>,
PC<15:13> → PCLATH<7:5>

Status Affected: None

Encoding:

110k	kkkk	kkkk	kkkk

Description: GOTO allows an unconditional branch anywhere within an 8K page boundary. The thirteen-bit immediate value is loaded into PC bits <12:0>. Then the upper eight bits of PC are loaded into PCLATH. GOTO is always a two-cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to PC
No operation	No operation	No operation	No operation

Example: GOTO THERE

After Instruction
PC = Address (THERE)

SLEEP **Enter SLEEP mode**

Syntax: [*label*] SLEEP

Operands: None

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

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

Encoding:

0000	0000	0000	0011

Description: The power-down status bit ($\overline{PD}$) is cleared. The time-out status bit ($\overline{TO}$) is set. Watchdog Timer and its post-scaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	Go to sleep

Example: `SLEEP`

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

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

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

SUBLW **Subtract WREG from Literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k – (WREG) → (WREG)

Status Affected: OV, C, DC, Z

Encoding:

1011	0010	kkkk	kkkk

Description: WREG is subtracted from the eight-bit literal 'k'. The result is placed in WREG.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example 1: `SUBLW 0x02`

Before Instruction
WREG = 1
C = ?

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

Example 2:

Before Instruction
WREG = 2
C = ?

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

Example 3:

Before Instruction
WREG = 3
C = ?

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

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**
D150	VOD	**Open Drain High Voltage**	–	–	8.5	V	RA2 and RA3 pins only pulled up to externally applied voltage
		Capacitive Loading Specs on Output Pins					
D100	COSC2	OSC2/CLKOUT pin	–	–	25	pF	In EC or RC osc modes, when OSC2 pin is outputting CLKOUT. External clock is used to drive OSC1.
D101	CIO	All I/O pins and OSC2 (in RC mode)	–	–	50	pF	
D102	CAD	System Interface Bus (PORTC, PORTD and PORTE)	–	–	50	pF	In Microprocessor or Extended Microcontroller mode
		Internal Program Memory Programming Specs (Note 4)					
D110	VPP	Voltage on MCLR/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 MCLR/VPP pin	–	25	50	mA	
D113	IDDP	Supply current during programming	–	–	30	mA	
D114	TPROG	Programming pulse width	100	–	1000	ms	Terminated via internal/ external interrupt or a RESET

† 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 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 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.

Note 1: When using the Table Write for internal programming, the device temperature must be less than 40°C.

2: For In-Circuit Serial Programming (ICSP™), refer to the device programming specification.

TABLE 20-18: A/D CONVERTER CHARACTERISTICS

Param. No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
A01	NR	Resolution		—	—	10	bit	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	10	bit	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A02	EABS	Absolute error		—	—	< ±1	LSb	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	< ±1	LSb	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A03	EIL	Integral linearity error		—	—	< ±1	LSb	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	< ±1	LSb	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A04	EDL	Differential linearity error		—	—	< ±1	LSb	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	< ±1	LSb	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A05	EFS	Full scale error		—	—	< ±1	LSb	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	< ±1	LSb	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A06	EOFF	Offset error		—	—	< ±1	LSb	VREF+ = VDD = 5.12V, VSS ≤ VAIN ≤ VREF+
				—	—	< ±1	LSb	(VREF+ — VREF-) ≥ 3.0V, VREF- ≤ VAIN ≤ VREF+
A10	—	Monotonicity		—	guaranteed(3)	—	—	VSS ≤ VAIN ≤ VREF
A20	VREF	Reference voltage (VREF+ — VREF-)		0V	—	—	V	VREF delta when changing voltage levels on VREF inputs
A20A				3V	—	—	V	Absolute minimum electrical spec. to ensure 10-bit accuracy
A21	VREF+	Reference voltage high		AVSS + 3.0V	—	AVDD + 0.3V	V	
A22	VREF-	Reference voltage low		Avss - 0.3V	—	AVDD - 3.0V	V	
A25	VAIN	Analog input voltage		AVSS - 0.3V	—	Vref + 0.3V	V	
A30	ZAIN	Recommended impedance of analog voltage source		—	—	10.0	kΩ	
A40	IAD	A/D conversion current (VDD)	PIC17**C**XXX	—	180	—	μA	Average current consumption when A/D is on **(Note 1)**
			PIC17**LC**XXX	—	90	—	μA	
A50	IREF	VREF input current **(Note 2)**		10	—	1000	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN
				—	—	10	μA	During A/D conversion cycle

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

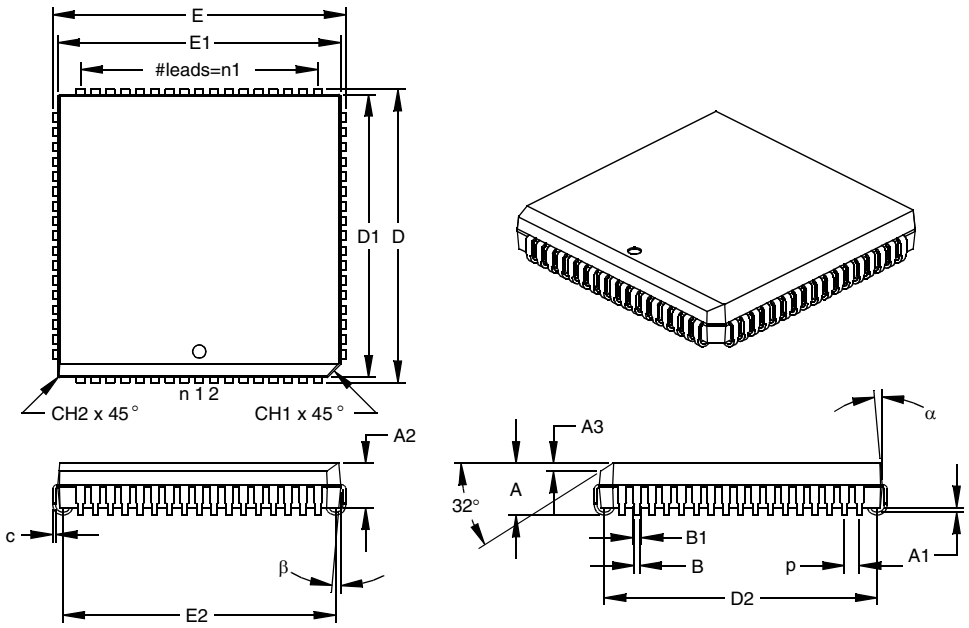
Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF current is from RG0 and RG1 pins or AVDD and AVSS pins, whichever is selected as reference input.

3: The A/D conversion result never decreases with an increase in the Input Voltage and has no missing codes.

68-Lead Plastic Leaded Chip Carrier (L) – Square (PLCC)

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		68			68	
Pitch	p		.050			1.27	
Pins per Side	n1		17			17	
Overall Height	A	.165	.173	.180	4.19	4.39	4.57
Molded Package Thickness	A2	.145	.153	.160	3.68	3.87	4.06
Standoff §	A1	.020	.028	.035	0.51	0.71	0.89
Side 1 Chamfer Height	A3	.024	.029	.034	0.61	0.74	0.86
Corner Chamfer 1	CH1	.040	.045	.050	1.02	1.14	1.27
Corner Chamfer (others)	CH2	.000	.005	.010	0.00	0.13	0.25
Overall Width	E	.985	.990	.995	25.02	25.15	25.27
Overall Length	D	.985	.990	.995	25.02	25.15	25.27
Molded Package Width	E1	.950	.954	.958	24.13	24.23	24.33
Molded Package Length	D1	.950	.954	.958	24.13	24.23	24.33
Footprint Width	E2	.890	.920	.930	22.61	23.37	23.62
Footprint Length	D2	.890	.920	.930	22.61	23.37	23.62
Lead Thickness	c	.008	.011	.013	0.20	0.27	0.33
Upper Lead Width	B1	.026	.029	.032	0.66	0.74	0.81
Lower Lead Width	B	.013	.020	.021	0.33	0.51	0.53
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

* Controlling Parameter
§ Significant Characteristic
Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MO-047
Drawing No. C04-049

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?