



constraint	
OSC	OSC = 20MHz
Period	Period = 0.2μs
	3
	1
	3
	1
	7
	3
	7

This use of **FSR** raises two conditions:

- If these I²C subroutines are executed from the mainline program, then any interrupt service routine that also uses **FSR** must set it aside upon entry and restore it upon exit.
- Any use of indirect addressing to access a sequence of addresses in the PIC's RAM when used in conjunction with these I²C subroutines must swap pointers in and out of **FSR**.

The timing requirements of Figure 9-6 will be handled by inserting a number of **nop** instructions between the instructions that change SDA and SCL. The number of **nop** instructions required depends on the crystal clock rate. The **delay** macro, defined in Figure 9-7a, uses the equate of **Freq** to 4, 10, or 20 to insert a number of **nop** instructions equal to the first, second, or third macro parameter.

The equates and variables needed for the I²C subroutines are listed in Figure 9-8. **DEVADD** is the selected peripheral chip's 7-bit address on the I²C bus shifted left one place to align it for use as a control byte. **INTADD** is a selected register or memory address inside the selected peripheral chip. **DATAOUT** is used to hold the byte of data to be sent to the selected register in the selected peripheral chip by an I²C output subroutine, **I2Cout**. **DATAIN** is the repository for the byte of data retrieved by an I²C input subroutine, **I2Cin**, from the selected register in the selected peripheral chip.

The **I2Cout** subroutine of Figure 9-9 calls a **Start** subroutine to generate the START condition, and calls a **TX** subroutine three times to send **DEVADD** (plus R/W = 0), **INTADD**, and **DATAOUT** out on the I²C bus. Finally, it calls a **Stop** subroutine to generate the STOP condition. The **TX** subroutine takes the byte passed to it in **W**, uses a **TXBUF** variable to extract the bits one by one, and transmits each bit using a **BitOut** subroutine. **TX** reads the acknowledge bit by calling a **BitIn** subroutine, setting **Z** if ACK occurs.

```

noexpand
delay macro freq4, freq10, freq20
  if Freq==4
    fill (nop),freq4
  endif
  if Freq==10
    fill (nop),freq10
  endif
  if Freq==20
    fill (nop),freq20
  endif
endm
    
```

(a) Macro definition

```

delay 0,1,2
    
```

(b) Example of macro invocation which will insert:

- 0 nop for OSC = 4 MHz (i.e., for Freq equ 4),
- 1 nop for OSC = 10 MHz (i.e., for Freq equ 10),
- 2 nops for OSC = 20 MHz (i.e., for Freq equ 20).

Figure 9-7 Delay macro.