İzmir Katip Çelebi University

Department of Electrical and Electronics Engineering

EEE316 Microprocessors

Spring 2020

Experiment II

Simulating I/O Ports, Branching and Loops

Pre-Lab Report

- Please study related topics in reference notes.
- Answer the questions under the lab activities. Prepare report in the specified format. Reports must be completed before coming to lab.
- Submit your report to CANVAS until March 4, 23:59.

Experimental Work

- Please explain your code step by step to instructors during lab hours.

Lab Objectives

- To examine the I/O port operation using a MPLAB simulator.
- To understand how to turn the LEDs on with given delay intervals.

References

- Lecture notes
- Mazidi, McKinlay, Causey "PIC Microcontroller and Embedded Systems," Chapter 3 and Chapter 4

Theory

There is two methods to generate "Delay" in PIC18F45K22 microcontroller :

1. Using TMR0, which is a built in timer in PIC18F45K22 microcontroller.
2. Using Delay loops Technique.

In this experiment, we will know how to generate a "Delay" with a certain value using the Delay Loop Technique; after that we will use this delay to make some applications. The Ordinary Instructions need 1 cycle to be executed, but the cycles which cause the Program Counter (PC) to be changed need 2 cycles.

Examples

For a PIC18 system of 4MHz,

4MHz ⟹ 1us

Instruction	Instruction Cycles	Time to execute
MOVWF PORTB	1 cycle	1x1 us = 1us
DECF MYREG	1 cycle	1x1 us = 1us
NOP	1 cycle	1x1 us = 1us
ADDLW	1 cycle	1x1 us = 1us
CALL DELAY	2 cycle	2x1 us = 1us
GOTO DELAY	2 cycle	2x1 us = 1us

Note : You can look for more instruction cycles to reference book's Appendix A.

Delay Subroutine

Consider the following delay subroutine

```
DELAY MOVLW D'255' ; outer loop is starting
      MOVWF 0CH
LOOP1  MOVLW D'255' ; intermediate loop starts
      MOVWF ODH
LOOP0 (1)NOP        ; inner loop starts
      .
      .
      (n)NOP
      DECFSZ 0DH,F
      GOTO LOOP0
      DECFSZ OCH,F
      GOTO LOOP1
      RETURN
      END
```

How to calculate the values of the Delay for this subroutine? (if n equals 13)

1. Look for the inner loop
   13(number of NOPs) + 1(DECFSZ 0DH,F)+ 2(GOTO LOOP0) = 16
2. Intermediate loop
   1(MOVLW .255) + 1(MOVLW 0DH)+ 1(DECFSZ 0CH,F)+2(GOTO LOOP1) = 5
3. Outer loop
   1(MOVLW .255) + 1(MOVLW 0CH)+ 2(RETURN) = 4

So, the value of the delay is (16*255*255+5*255+4)*(4/osc freq) = 1.04sec (if osc freq = 4MHz)

Note that this value of the delay is not precise 100%, since we ignore that the instruction DECFSZ needs 2 cycles in the last turn, and hence this method (Delay Loops) is not effective for precise and large values of Delay.

Lab Activities

1. **(MPLAB and PROTEUS)** Write a program increment from 00 to D'40' and send to PORT C (Value should send to PORT C after each increment). Use once "CALL" subroutine for time delay. Put a time delay 0.8s in between each issuing of data to PORT C. Examine the code on PROTEUS, connect 8 LEDS to PORT C and observe time delay.

2. **(MPLAB and PROTEUS)** Write and assemble a program to toggle all the bits of PORTB, PORTC and PORTD continuously by sending 55H and AAH to these ports. Put a time delay of 1.2s between the "on" and "off" states given that your internal clock frequency is 16MHz. Using mpasm assembler, build your code and obtain the HEX file output. Examine the code on PROTEUS and observe the LED blinking. Find the size of delay routine in this programme for clock of 16MHz. Write your result to your pre-lab report.

3. **(MPLAB and PROTEUS)** A shop window lighting system has two different LEDs. One of them is red and other one is yellow. When shop manager press the button, red LED turns on, and turns off after 0.5s. When red LED turns off, yellow LED turns on and turns off after 0.5s. Then red LED turns on and it should continue in the same way (red-yellow-red-yellow…). Red LED is connected to RB0, yellow LED is connected to RB1 and the button is connected to RC0. Write this program and simulate the program via PROTEUS software. Verify it operates properly when simulated.