Homework 2

Due: Sep. 9, 2015 11:30pm

Objective

Learn simple HC12 assembly program for the parallel port I/O, subroutine, looping, and timing.

Instruction

1. Read the textbook Chapter 1, Chapter 2, Chapter 3, and Chapter 4. Note, you do NOT need to understand all of them at this time.

2. Be sure to do the HW1 if you have not done it yet. You should be familiar with the assembly programming process from the HW1 and HW2. That is,
(1) start the Freescale CodeWarrior IDE,
(2) do 'New Project' named hw2,
(3) type in the source code,
(4) click the 'Make' icon to assemble and generate the .s19 executable file,
(5) click the 'Debug' icon to start the Full Chip Simulation,
(6) add I/O components, and
(7) verify correct program functioning or do debug the program.

3. Create HW2 in CodeWarrior, type in the sample program text from hw2prg.pdf file. Do **make** the main.asm file and generate the .s19 executable file.

4. Once the program has been assembled without error, Start the Full Chip Simulation Debug.

5. Add components: 1 io_LED and 1 Push_Button to simulator.
Set io_LED to PORTB: Port=1 and DDR=3.
Set Push_Button to PORTP: Port=259.
(Read the debugging aid section below for details.)

6. Run or single step your program. Be sure that the PC is set to $3100 on your simulator upon start.

7. When you are running the program, click left mouse button over the Push_Buttons switch bit 0 and see the effects on the LED 1 at bit 4. The speed of the LED 4 at bit 7 blinking can be changed. It may be too slow or it may be too fast.

8. The Push_Button switch bit 0 is the SW1 (Switch 1) on the CSM-12C128 board. Now observe the LED blinking, press SW1 and see the LED light.

9. Try to understand the program.

10. Try changing the delay values and observe the LED blinking speed.

11. Modify the program including the comments. Have your program blink LED lights on

PORTB7 and PORTB4 alternately. Use SW1 input to change the blinking pattern to another (go wild).

12. In another words, the 'blink LED on 7 and 4 alternatively' mean that both of the lights should blink at the same time - while 7 is ON and 4 is OFF for 1 second and then 7 is OFF and 4 is ON for next 1 second, and repeat.

13. Also what is SW1 used for? Well, if sw1 is **NOT pressed**, do the alternat bliking above. If sw1 is **pressed**, do something different than the alternate blinking above. For example, blink all 4 LEDs in sequence: LED7 ON and others OFF, LED6 ON and others OFF, LED5 ON and others OFF, LED4 ON and others OFF, and repeat. There are many other possibility with LEDs.

14. Again assemble, simulate, and run your program on the Full Chip Simulator.

15. Be sure to correct all the errors and be sure to update the comments (put your name, etc.). Comments are very important in assembly language programs.

16. You may want to see and check the Sample Grading Sheet for this homework.

17. Copy your 'main.asm' file to 'cmpen472hw2_YourLastName.asm'. For example, mine will be 'cmpen472hw2_choi.asm' (Do not ZIP your 'cmpen472hw2_YourLastName.asm' file.)

18. Turn-in your project source code asm file through Penn State ANGEL. Deposit your source code asm file into the Homework 2 DropBox under CLASS tab in CMPEN 472 Course.

Congratulations on your second CMPEN 472 homework completion!

Debugging Aid

1. You may use debugger if your program is not working - you need to debug.

2. Start CodeWarrior debugger. On the CodeWarrior Debugger top line pull-down menu, find 'Component' and 'Open' io_LED and Push_button. Set up the components as follow: io_LED at PORT = 1 (PORTB) and DDR = 3, Push_button at PORT = 259 (PORTP).

3. Run the program in the debugger, observe PORTB7 LED blinking. If you press the push button bit 0, PORTB4 LED will turn off. If you release the push button bit 0, PORTB4 LED will be turned ON.

4. Change the delay time on the program to blink faster in your PC. When you change the program source file, you need to re- **Make** the main.asm file and restart the debugger.

5. If you set the delay time on the program as minimum, then you can do single step through the program to check the LED light ON and OFF. Again when you change the program source file, you need to re- **Make** the main.asm file and restart the debugger.