

ELEC 2220 Computer Systems
Homework #14
Due: Wednesday, July 7

NOTE: The remaining homework projects for this semester require CodeWarrior version 4.6 or 4.5 (or earlier). The newest download on Freescale's web site, version 4.7, has eliminated the simulated components that we need for our projects. The CD in the text book contains version 4.5, and that the labs in Broun Hall have either version 4.5 or 4.6. So, if you downloaded version 4.7 or later, you will either need to use a PC in the lab, or install version 4.5 from the text book CD.

Code Warrior provides a number of simulated components that can be connected to the simulated microcontroller. In this exercise, you will work with parallel input/output ports by reading the states of 8 push buttons, and displaying the number of any depressed button on a seven-segment LED display.

Read sections 18.2 and 18.3 of the text book, and in particular examine Figure 18-7. A 7-segment LED display is to be connected to Ports A and B of the HCS12. The bits of Port A are to be connected to the seven LED "segments" a-g (segments activation), with bit 0 of Port A activating LED segment "a", bit 1 of Port A activating LED segment "b", etc. (A logic 1 activates the LED segment.) Port B is to be connected to the "display selection" signals of the 8-digit display component. Forcing bit 0 of Port B to logic 1 causes the segments of the right-most digit to display the data from Port A, bit 1 of Port B selects the second digit from the right, etc.

To create this display in the CodeWarrior debugger, in the debugger menu bar select Component > Open. This opens a window showing a set of "components" that can be used. Scroll down and select the "Segments" component. This opens an 8-digit display component. Right-click on the component and then click on "setup". Enter the HCS12 address of Port A in the "Segments activation" box, and the address of Port B in the "Select a display" box. (See text chapter 11 for port addresses.)

The 8 push buttons are to be connected to the pins of Port T. Create this in the debugger by selecting a "Push Buttons" component from the Component menu. Right-click on the component, click on "setup", and enter the address of Port T in the box. This connects button 0 to bit 0 of Port T, button 1 to pin 1, etc.

The display should initially be blank (initialize in your program). Your program is to read the states of the buttons, and if any button is found to be pressed (logic 1), display the button number (0, 1, 2, 3, 4, 5, 6, 7) on the right-most digit of the display. That number should continue to be displayed until another button is pushed.

Submit your source program and a screen capture of the debug window, showing the 7-segment display, with the display showing the number of the last button pushed.