ECE-3120
Fall 2008

LAB 4 – Min/Max Program

The purpose of this lab is to reinforce your basic programming skills with the 68HCS12 using program loops, arrays, comparisons, and arithmetic instructions.

PRE-LAB:

Prepare pseudocode and the first draft of the program and calculate the expected results by hand (both decimal and hex). This must be completed before coming to the lab and shown to the lab instructor at the start of the lab session. Note: The Pre-Lab must be typed into a proper *.ASM source file, following our standard Program Format requirements.

Approved: Lab TA ______________________ *Date* ____________

PROGRAMMING ASSIGNMENT:

Write a fully-commented program for the HCS12 board, including appropriate directives and labels for memory operands and constants, called **Minmax.asm**. It must use a program loop and comparisons for most of the work. The program should do the following:

- Be able to process any array of 1 to 255 signed data words as follows. Each data word can have a value between -32768 and +32767. This means that the program should be able to handle any size array in this range, with any set of values, by changing ONLY the directive that creates the array values, without modifying any instructions or other directives.
- The size of the array cannot be stored as a pre-defined constant. A zero value in the array is a delimiter (special flag value) that means that you have reached the end of the array, so stop all processing at this point. This is another way to terminate a loop, without using a fixed number inside your program (or in a directive). The program must not include the delimiter of zero in calculating any of the numerical results.
- Count the number of array elements that are exact multiples of 5.
- Search through the array, looking for the maximum (most positive) and minimum (most negative) numbers. Also give the array index of the maximum number.
- The code must start at $1000.

The program directives should initialize the following **signed decimal data word array** in the data space (starting at $1200):
`10,-32768,524,400,3,-234,55,8000,32767,0`
in this order.

- The program should store in memory:
- The signed minimum value in memory location $1300:$1301
- The signed maximum value in memory location $1302:$1303
- The index of the maximum value in memory location $1304
- The count of values that are multiples of 5 in memory location $1305

Procedure:
First, use D-Bug12 to fill memory locations $1000 through $3BFF with zeros. Then assemble, download, and debug/execute the program as follows.

1. Set a breakpoint to stop execution at the end of each loop and run through the program to the end, pausing at each breakpoint to display the important values. Verify that each value is correct at the end of each loop and that all final results are correct at the end of the program. Use the listing file to determine the breakpoint address. Remember that the instruction at the breakpoint address is NOT executed when the break occurs; it is only executed when you resume execution after the break.
2. Then reset the processor (which removes the breakpoint), download the program again, run it at full speed until it stops, and verify that the four final values are still correct.
3. When finished debugging and executing, copy the entire terminal window output and paste it into a Notepad or Word document for inclusion in the report. You should edit out mistakes and unnecessary repetitions before submission.

Approved: Lab TA ______________________ *Date* _____________

Things to turn in as your Lab Report, attached in this order:

1. This assignment sheet, with your name at the top, signed by the TA where shown.

2. Your uncorrected pre-lab document (commented source code).

3. A printout of the final **Minmax.asm and Minmax.lst** files. You'll need to print the listing file in landscape mode to make it fit. Use Notepad to print them.

4. A printout from the terminal screen (method: highlight text, type ctrl+c, and paste into a Notepad or Word document, using Courier New as the font) that includes everything done. Add brief hand-written comments and highlight important values in the procedure, showing the relevant memory contents and register contents.

5. **Answer the following questions, in your document:**

    a. What are the final values, in both hex and decimal? Label each value clearly.

    b. What are the advantages and disadvantages of having a list delimiter (special code) to indicate the end of an array, instead of using a predefined constant for the array size?

    c. Why do we use the block-fill command with all zeroes?