Jacob Cox
9/13/2017

EE 3171 Lab 1 Submission

Code Entry and Debugging

1 Submission Requirements

Answer every question. Make sure that if you use any resource not provided to you by the TA, instructor or in the textbook you properly cite it. All demonstration requirements have a matrix to be filled out by the TA. This will allow partial credit to be given and for it to be clear how things are graded. Any demonstration required, you need to ask a TA to grade it. Only authorized individuals can grade your demonstrations. You **must** submit your code for all programs. If code is not submitted then you will lose the demonstration points.

2 Code Entry (50 points)

1. After the initial build of the unedited code, indicate how you fixed:
   - (10) The warning on/around Line 28

     added # before the 5
   - (10) The easy ~~warning~~ error

     needed a space before ADD on line 13
   - (15) The "terrifying error" (include all the lines that reference k)

     needed an = before K on line 6

2. (5) If you encountered any other errors or warnings, list them here.

the only other warnings/errors I had were typos

3. (5) The initial values for `j` and `k` are both assigned in the section of code labeled `SystemInit`. What are those initial values?

j = 5

k = 0x3

4. (5) The three variables are all assigned using different bases and specifications. Did this have any effect on the way the code worked? Why or why not?

No. They are just different representations of numbers. Different ways to say the same thing.

3 Simulation (100 points)

1. (50) Demonstrate to the TA that your program successfully builds and runs.
   TA Initials: H.S.
   Date: 09-13-2017

2. (5) What is the value of the PC when the debug session is started?

0x01000000

3. (5) Which of the two files does your project start in?

the startup

4. (5) What is the value of the PC and what is the instruction that causes the value of `theAnswer` to change in the Watch Window?

PC = 0x00000 3A0

16 STR R5, [R6]

5. (5) What is the value of R4 that finally causes the loop to exit in the initial version of the code?

0x00000003

6. (5) In the initial version of the code, how many times is the code on line 14 executed (the ADD R4, R4, #1 line)?

Three times

7. (5) What was the final value for `theAnswer` in the initial version of the code?

0x0000000F

8. (20) Based on your reading of the code (and other values you used to test your hypothesis), what does this program do?

The end result is the variables (j and k) being multiplied together.

The program is adding j k-amount of times in the "loop". The label R4 is being used as the loop counter, and it's being compared to the total loops R3. Once R3=k the loop braces off and the temporary stored value R5 is stored in the variable theAnswer. Then the program loops forever in the black hole.