

CPE 325: Embedded Systems Laboratory

Laboratory Assignment #5

Assignment

[100 pts]

1. Write an assembly program that passes a base number b (value should be other than 0 and 1) to a subroutine `calc_power`. This subroutine should populate two arrays in memory with b^1, b^2, b^3, b^4 and b^5 . This means your subroutine should compute first 5 powers of a parameter passed into it. *[One of the arrays is populated with results using hardware multiplier and the other using software multiplier.]*

You must pass b to `calc_power` using a register of your choice. You may also want to pass the address of your result. Pass these addresses using stack.

In order to compute the powers, you need multiplication operations. For this you must implement two additional sub routines as defined below in Q2.

2. One of the subroutines that you need to implement is `SW_Mult` that uses Shift-and-Add multiplication algorithm. The algorithm is described in provided pdf file. Another subroutine that you need to implement is `HW_Mult`. It should use Hardware Multiplier to multiply numbers.

Both of these subroutines should take in input numbers through stack and return the result of multiplication in one of the registers.

Hints:

- a) *You may want to allocate space for results in main. You can allocate space using `."bss"` for both results using hardware multiplier and software multiplier.*
 - b) *In subroutine `calc_power`, you can repeatedly call SW multiplier subroutine to populate the memory space allocated for software based result. Then, you can use the similar approach to populate memory space allocated for hardware based result.*
3. Measure the number of clock cycles used by each subroutine for a small range of values. Comment of the efficiency of each subroutine.
 4. **Bonus (up to 15pts)** Create a subroutine that converts a string of at least a five-digit number to its numerical value by using the Hardware Multiplier and stores the result in a variable in the memory. The subroutine needs to receive the base address of the string and the address of the variable through the program stack. For full credit you need to use the accumulator. If the accumulator is not used only up to 10pts will be rewarded.

Questions To Be Addressed

Please make sure that you have addressed following questions in your demonstration:

1. How do you pass parameters to a subroutine using stack? Explain how you extract parameters that you pass using this technique.

Topics For Theory

1. Subroutines
2. Passing parameters (3 different ways)
3. Hardware Multiplier

Deliverables

1. Lab report with screenshots of final outputs
2. Commentary on efficiency of each subroutine (in terms of clock cycles taken for operation)
3. Source files (.asm files) or as instructed.

Notes:

1. Try different inputs before you conclude which method is efficient. In your explanation, include the inputs as well.
2. Assume that any of the results does not exceed 16-bits.