Pratik Gangwani

ECE 3056

Lab 1 Extra Credit Report

Results:



The avg. # of instructions to perform MUL is: 3001/10 = 300.1 or ~ 301

To start the assignment I looked at the byteadd.asm provided to us to get back into writing assembly code. Reading through the code I realized that much of it was useful to our own assignment for writing a MUL function. As such, much of the beginning and end parts are the same as byteadd.asm; mainly the storing of the answer at the end, and retrieving the inputs at the beginning.

I chose to go with the repeated addition method to write the ‘mul’ function. It is inefficient compared to shifting the bits, but was much easier to implement and debug. The first time I wrote the function I used a counter and flipped it’s bits and added 1 to get the value negative one, while repeatedly adding the first input to it self. I ended up having 393,234 instructions for 0x01 \* 0xFF! I then rewrote the code using the second input minus a constant of negative one to control my loop and added in branches to check for whether the inputs themselves were zero. This cut my instructions in half and brought 0x01 \* 0xFF to 19 instructions.