ECE 3221 Lab 2

Alexis Savoie 3389902

1. We were able to check if any keys were being pressed as there are specific values that correspond to no keys being pressed. By having this value, we were able to constantly scan to see if the value was changing or not. If it had changed, that means a key had been pressed.
2. In part 2, the program was then modified to only send the code to the hex display if a key was being pressed all while only the first row is selected.
3. The program was then modified to scan all the rows to see if a key was pressed.
4. The lookup table containing the hex digits corresponding was then added to our code. The first address of these stored scancodes was 0x300. To read the first byte of the scan code, the address of MYTAB was placed into a register. This was then used to retrieve the actual values of the hex digits. Labels were then used to compare the value of the hex digit to the value of the key being pressed. If the values did not match the value of the of the key pressed, the current stored address of the hex digit was incremented, using the addi function, to move to the next address. We then branched back to the top of our label to compare them again. If the values did match we wrote the hex digit to the 7-segment display.
5. If a scancode value from the input does not match one found in the lookup table, the program will continue searching in the memory until a correct value is found. If no value is found it is possible we may loop forever. However, I do not expect this to happen as all the keys on our control register have been accounted for unless two values are pressed at the same time. If two values are pressed at the same time the program will no longer work or display anything as there is currently nothing in our table or memory that would match the value of two buttons being pressed at the same time. This is why “.byte 0X00, 0XFF” is added to the lookup table so that we can display FF is this occurs.
6. A scrolling display was then added to our code by using some of the code that was provided. When a key was pressed it was displayed in the first 7 segment display while all the values stored before this value were shifted to the 7 segment displays to the left.
7. For part 7 the code provided to us was added to our program. This allowed us to ensure that a key was pressed firmly and that all the bounce has passed.