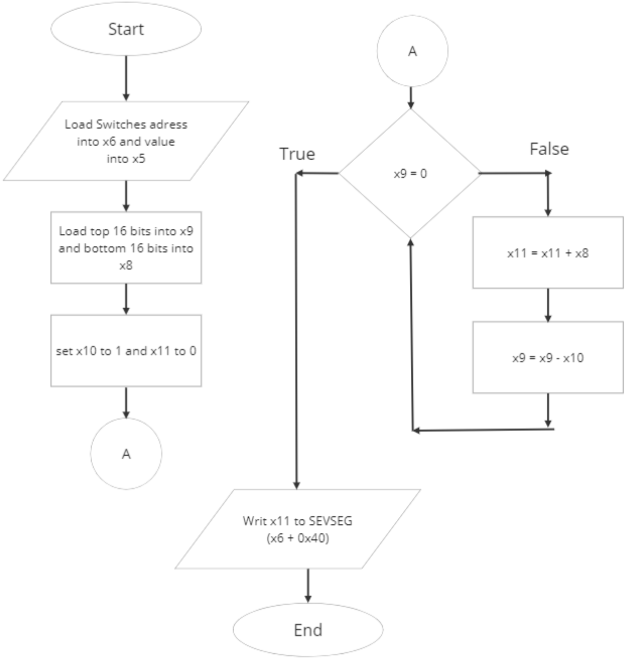
CPE 233 SW 3

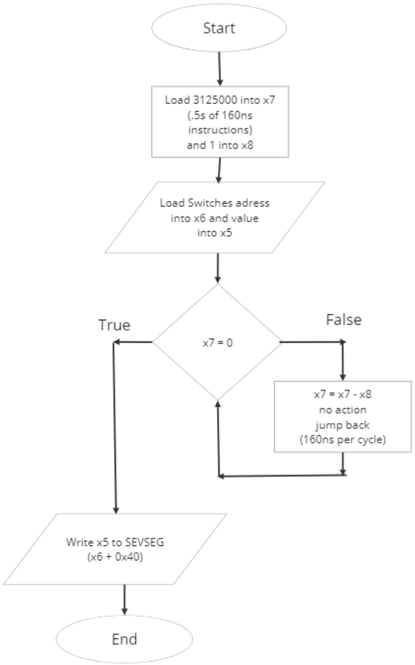
Wyatt Tack

1. Flowcharts :

Part 1 :


Start
Load Switches adress into x6 and value into x5
Load top 16 bits into x9 and bottom 16 bits into x8
set x10 to 1 and x11 to 0
A
A
x9 = 0
True
False
x11 = x11 + x8
x9 = x9 - x10
Writ x11 to SEVSEG (x6 + 0x40)
End

Part 2 :


Start
Load 3125000 into x7 (.5s of 160ns instructions) and 1 into x8
Load Switches adress into x6 and value into x5
x7 = 0
True
False
x7 = x7 - x8 no action jump back (160ns per cycle)
Write x5 to SEVSEG (x6 + 0x40)
End

2. Table 1: Simulation Table Part 1:

Reasoning	Switches	SevSeg Memory
Tests standard easy multiplication	**8 x 4** **0x0008_0004**	**32** 0x0000_0020
Tests maximum	**65535 x 65535** **0xffff_ffff**	**4294836225** 0xfffe0001
Tests 0	**0** **0x00000000**	**0** 0x00000000
Another standard high digit multiplication test	**4667 x 53835** **0x123b_d24b**	**251247945** 0x0ef9bd49
One more simple multiplication test	**15 x 5** **0x000f_0005**	**75** 0x0000004b

Table 2: Verification Part 2:

Reasoning
To achieve 0.5s at each instruction running at 40ns, a total of 0.5/(40*10^{-9}) = 12500000 total instructions are needed between loading the value and outputting the value. To achieve this, we need at least 3 commands per loop (the conditional, the loop counter, and the jump to the beginning of the loop). A fourth "no operation" command was added to exactly be able to divide the number of total instructions by 4, instead of by 3. This 4 operation loop at 40ns per operation makes a time of 160ns per loop, leading us to need 3125000 loops total

3. Figure 1: Assembly Code Part 1:

```
        lui x6, 0x11000 #set x6 as value for Switches memory address
        lw x5, 0(x6) #fill x5 with value in switches
        slli x8, x5, 16
        srli x8, x8, 16 #shifts 16 bottom bits into x8
        srli x9, x5, 16 #shifts 16 top bits into x9
        lui x11, 0 #initializes storage for multiplication
        addi x10, x0, 1 #adds value for subtraction in loop
LOOP:   beq x9, x0, END #add x8 into x11 for every value of x9
        add x11, x11, x8 #definition of multiplication
        sub x9, x9, x10
        j LOOP
END:    sw x11, 0x40(x6) #load into sevseg
```

Figure 2: Assembly Code Part 2:

```
        li x7, 3125000 #.5s/160ns = 3125000
        addi x8, x0, 1
        lui x6, 0x11000 #set x6 as value for Switches memory address
        lw x5, 0(x6) #fill x5 with value in switches
LOOP:   beq x7, x0, END #160ns loop, counts x7 ammounts of times
        sub x7, x7, x8
        nop             #160ns * 3125000 = 0.5s
        j LOOP
END:    sw x5, 0x40(x6) #load into sevseg
```