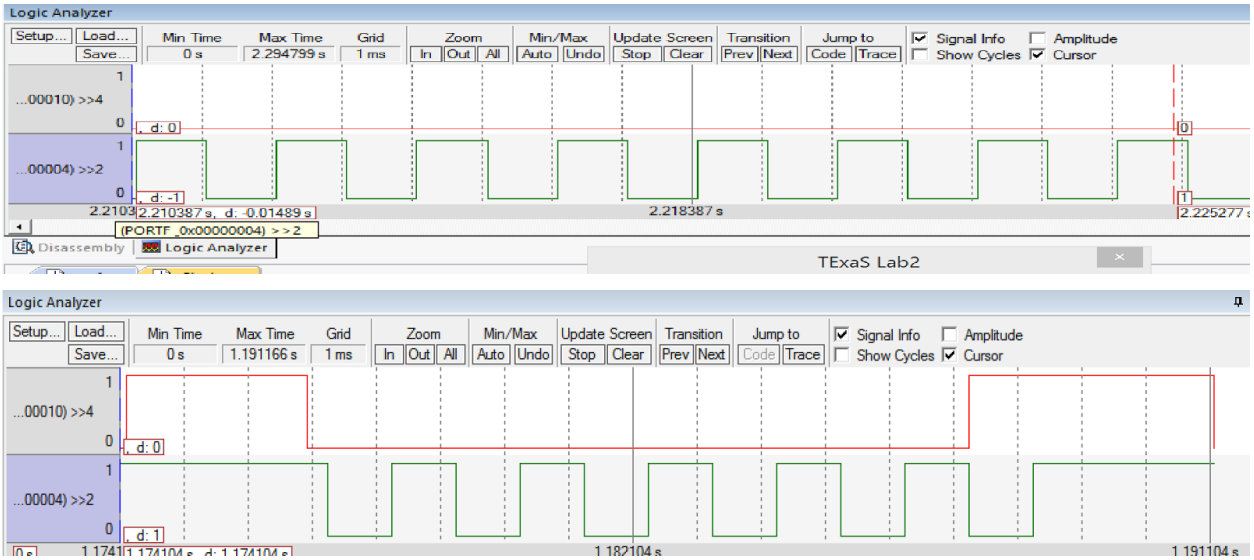
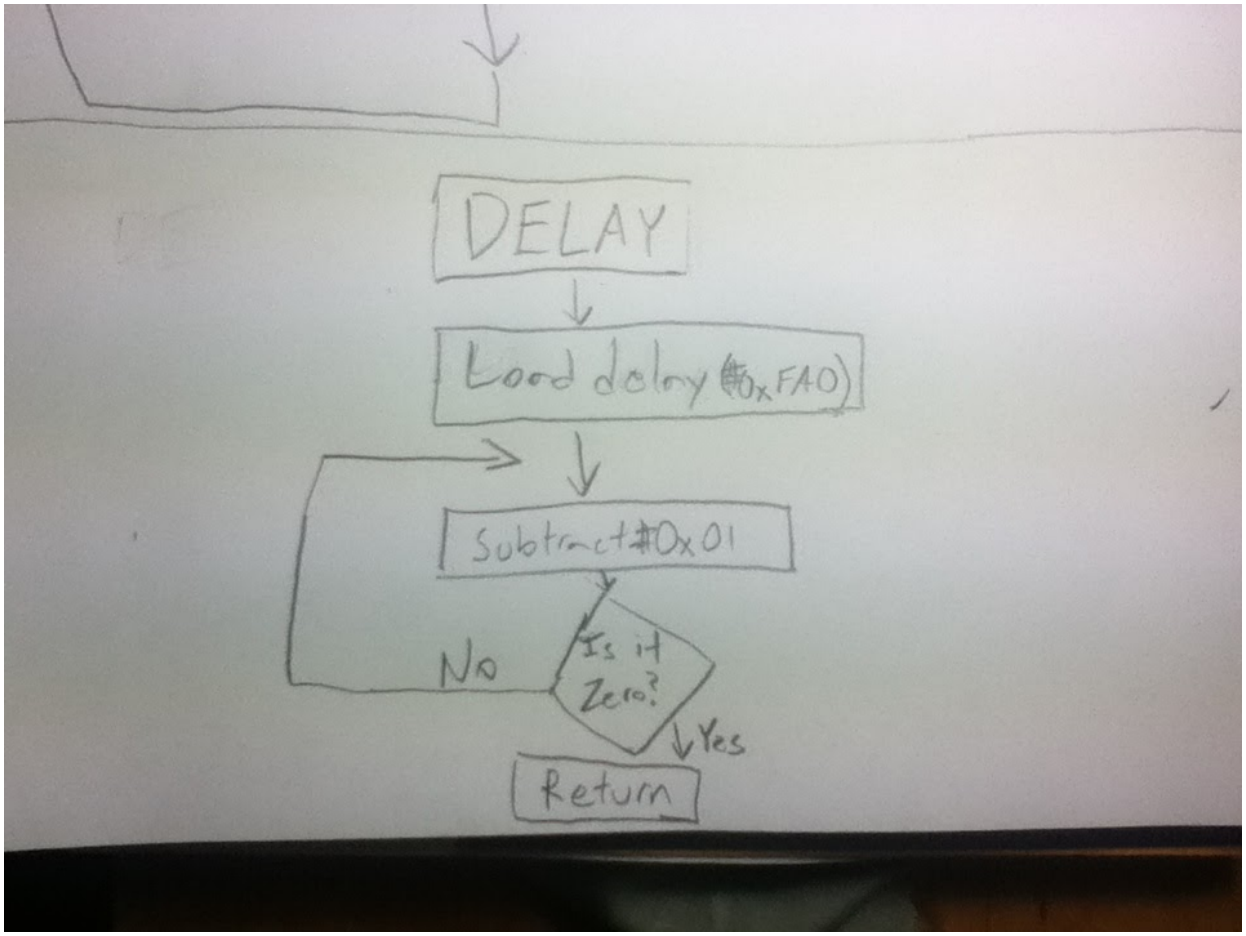
Briar Coker
Kyle Sikora

EE 319K Lab 2

Logic Analyzer:



Delay Flowchart:



Delay Pseudocode:

```
Delay       Load #0xFA0 into R0          ;1ms
Wait        Subtract #0x01 from R0
            Check if R0 is zero          ;counts all the way down
            If R0 is not zero branch to wait
            Else RETURN                  ;done
```

Assembly Source Code:

```
GPIO_PORTF_DATA_R      EQU   0x400253FC
GPIO_PORTF_DIR_R       EQU   0x40025400
GPIO_PORTF_AFSEL_R     EQU   0x40025420
GPIO_PORTF_PUR_R       EQU   0x40025510
GPIO_PORTF_DEN_R       EQU   0x4002551C
GPIO_PORTF_AMSEL_R     EQU   0x40025528
GPIO_PORTF_PCTL_R      EQU   0x4002552C
SYSCTL_RCGCGPIO_R      EQU   0x400FE608

      AREA    |.text|, CODE, READONLY, ALIGN=2
      THUMB
      EXPORT  Start
Start
        BL init
loop
        BL delay
        LDR R0,[R1];
        ANDS R0,#0x10 ;ANDS is needed to set NPVC flags
        BNE TurnLED_ON    ;Branches to TurnLED_ON if switch is not pressed
ToggleLED                 ;continues to ToggleLED if switch is pressed
        LDR R0,[R1]
        EOR R0,#0x04      ;Exclusive OR will toggle LED in DATA every time it executes
        STR R0,[R1]
   B loop
TurnLED_ON
        LDR R0,[R1];
        ORR R0,#0x04
        STR R0,[R1];
        B loop
delay
        MOV R0,#0xFA0     ;1 Cycle, 0xFA0 is approximately 1ms
wait
```

```
        SUBS R0,#0x01       ;1 Cycle, counts down
        BNE wait            ;(1 or 1 + p) [Average 3 Cycles]
        BX LR;
init
;Turn on Port F Clock
        LDR R1, = SYSCTL_RCGCGPIO_R;
        LDR R0, [R1];
        ORR R0,#0x20;
        STR R0,[R1];
        NOP
        NOP
;Set Pin Directions
        LDR R1, = GPIO_PORTF_DIR_R;
        LDR R0, [R1];
        BIC R0,#0x10;
        ORR R0,#0x04;
        STR R0,[R1];
;Turn off Alternate Functions
        LDR R1, = GPIO_PORTF_AFSEL_R;
        LDR R0, [R1];
        BIC R0,#0x14;
        STR R0,[R1];
;Enable Digital Pins
        LDR R1, = GPIO_PORTF_DEN_R;
        LDR R0, [R1];
        ORR R0,#0x14;
        STR R0,[R1];
;PULL UP RESISTOR ENABLED!!!
        LDR R1, = GPIO_PORTF_PUR_R
        LDR R0,[R0]
        ORR R0,#0x10
        BIC R0,#0x04
        STR R0,[R1]
;SET PF2 =1 System starts like this
        LDR R1, = GPIO_PORTF_DATA_R;
        LDR R0,[R1];
        ORR R0,#0x04;
        STR R0,[R1];
        BX LR

    ALIGN      ; make sure the end of this section is aligned
    END        ; end of file
```

Time Comparison:

10 seconds in real time translated to 3.4 seconds in the simulation. This is most likely an effect of using bootcamp to run windows on a macbook, which uses a significant amount of processing power.