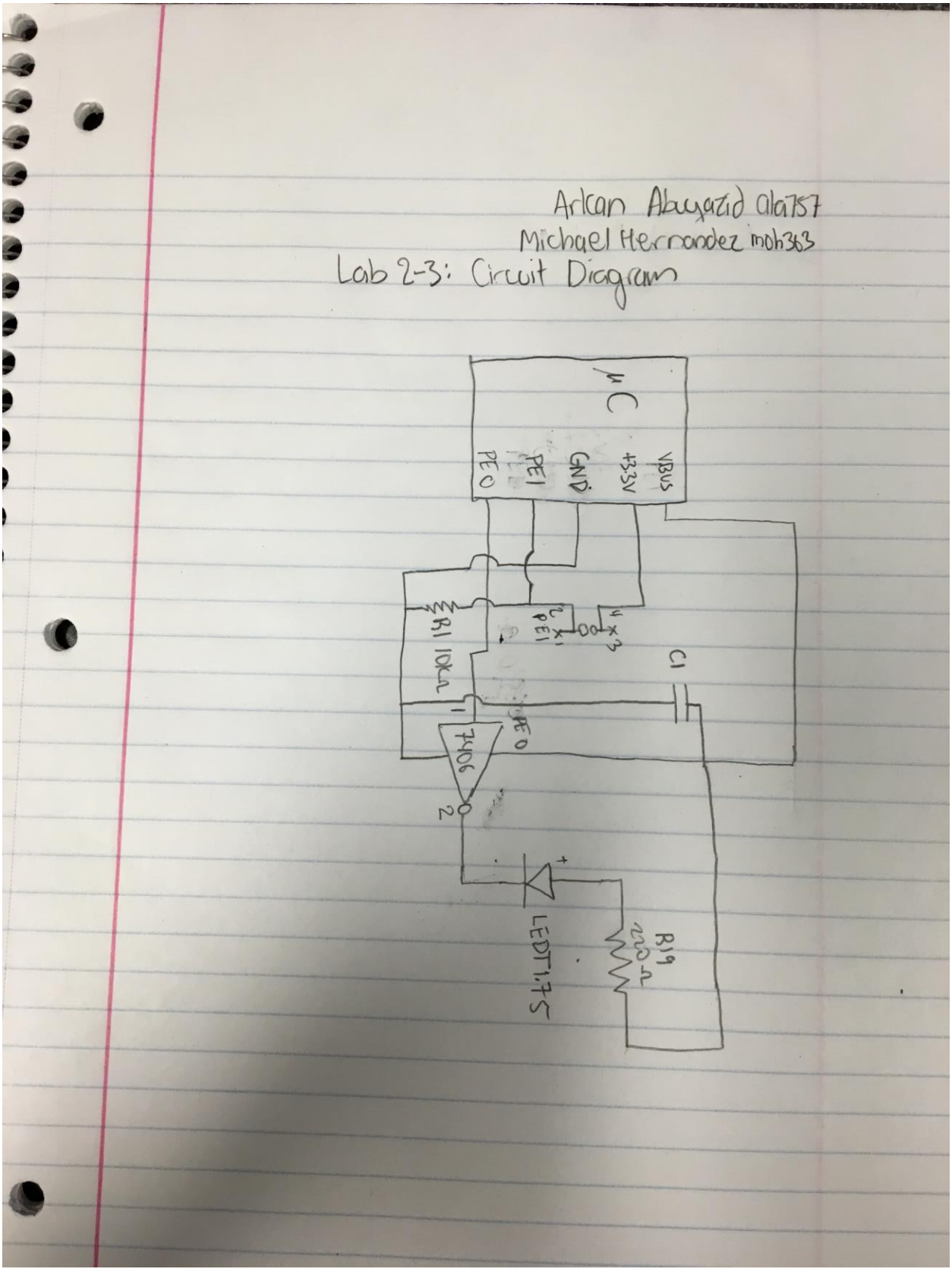
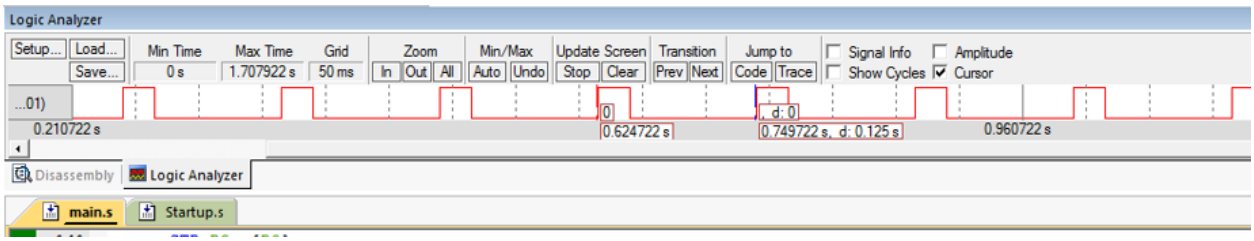
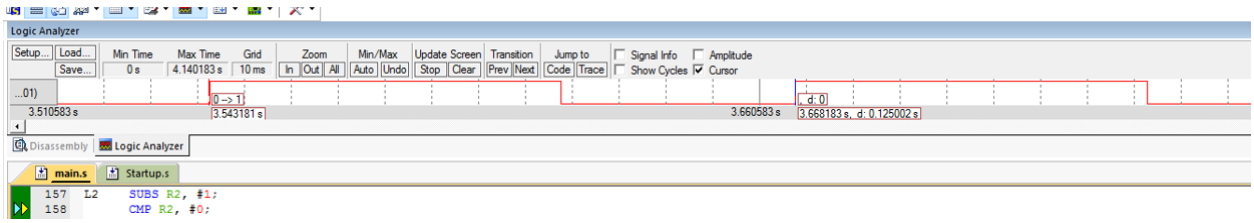
2.) CIRCUIT DIAGRAM



3.)


Logic Analyzer
Setup...
Load...
Save...
Min Time
0 s
Max Time
1.707922 s
Grid
50 ms
Zoom
In
Out
All
Min/Max
Auto
Undo
Update Screen
Stop
Clear
Transition
Prev
Next
Jump to
Code
Trace
Signal Info
Show Cycles
Amplitude
Cursor
...01)
0.210722 s
0
0.624722 s
d: 0
0.749722 s, d: 0.125 s
0.960722 s
Disassembly
Logic Analyzer
main.s
Startup.s


Logic Analyzer
Setup...
Load...
Save...
Min Time
0 s
Max Time
4.140183 s
Grid
10 ms
Zoom
In
Out
All
Min/Max
Auto
Undo
Update Screen
Stop
Clear
Transition
Prev
Next
Jump to
Code
Trace
Signal Info
Show Cycles
Amplitude
Cursor
...01)
0 -> 1
3.510583 s
3.543181 s
d: 0
3.660583 s
3.668183 s, d: 0.125002 s
Disassembly
Logic Analyzer
main.s
Startup.s
157 L2 SUBS R2, #1;
158 CMP R2, #0;

4.)

Arkan Abuyazid ata757
Michael Hernandez moh363

Lab 2-3: Switch Measurements

Parameter	Value	Units
Resistance of 10k.Ω, R1	9880	Ω
Supply Voltage V+3.3V	3.295	V (Volts)
Input Voltage V_{PE1}	0.00	V (Volts)
Resistor Current	0.00	mA (Amperes)
Input Voltage V_{PE1}	3.295	V (Volts)
Resistor Current	0.33	mA (Amperes)

5.)

Lab 2-3: LED Measurements

Parameter	Value	Unit
Resistance of 220Ω Resistor, R19	218.2	Ω (Ohms)
+5V power supply V_{+5V}	5.65	V (Volt)
TM4C123 Output V_{PE0}, input to 7406	88.4	mV (Volts)
7406 Output V_{K-}, LED_{K-}	4.10	V (Volts)
LED_{A+} V_{A+}	5.38	V (Volts)
LED Voltage	1.28	V (Volts)
LED Current	0.00	mA (Amperes)
TM4C123 Output V_{PE0}, input to 7406	3.289	V (Volts)
7406 Output V_{K-}, LED_{K-}	131.4	mV (Volts)
LED_{A+} V_{A+}	1.930	V (Volts)
LED Voltage	1.798	V (Volts)
LED Current	13.72	mA (Amperes)

6.) Assembly code

```
GPIO_PORTE_DATA_R  EQU 0x400243FC
GPIO_PORTE_DIR_R   EQU 0x40024400
GPIO_PORTE_AFSEL_R EQU 0x40024420
GPIO_PORTE_DEN_R   EQU 0x4002451C
; PortF device registers
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
TERM               EQU 0x0002867C; one millisecond
SYSCTL_RCGCGPIO_R  EQU 0x400FE608


      AREA    DATA, ALIGN=2


CYCLE SPACE 4; duty cycle percent,
Blink SPACE 4;DUTY CYCLE FOR BREATHE
COUNT SPACE 4; I don't think I need this
BCO2  SPACE 4; BREATHE COUNT

                    ALIGN      ; make sure the end of this section is aligned


      IMPORT  TExaS_Init
      AREA    |.text|, CODE, READONLY, ALIGN=2
      THUMB
      EXPORT  Start
```

Start

```
LDR R1, =SYSCTL_RCGCGPIO_R; TExaS_Init sets bus clock at 80 MHz
LDR R0, [R1];Initialization
ORR R0, R0, #0x30;
STR R0, [R1];
NOP;
NOP;
NOP;
NOP;
LDR R1, =GPIO_PORTE_DIR_R;
LDR R0, [R1];
AND R0, R0, #0xFD;
ORR R0, R0, #0x01; PE1 input and PE0 output
STR R0, [R1];
LDR R1, =GPIO_PORTE_AFSEL_R;
LDR R0, [R1];
AND R0, R0, #0x00;
STR R0, [R1];
LDR R1, =GPIO_PORTE_DEN_R;
LDR R0, [R1];
```

```
        ORR R0, R0, #0x03; enables just PE0 and PE1
        STR R0, [R1];
        LDR R1, =GPIO_PORTF_DIR_R;
        LDR R0, [R1];
        ORR R0, R0, #0x00; PF4 INPUT
        STR R0, [R1];
        LDR R1, =GPIO_PORTF_AFSEL_R;
        LDR R0, [R1];
        AND R0, R0, #0x0000;
        STR R0, [R1];
        LDR R1, =GPIO_PORTF_PUR_R;
        LDR R0, [R1];
        ORR R0, R0, #0x10;
        STR R0, [R1];
        LDR R1, =GPIO_PORTF_DEN_R;
        LDR R0, [R1];
        ORR R0, R0, #0x10;
        STR R0, [R1];END of initialization

        ; TExaS_Init sets bus clock at 80 MHz
    BL  TExaS_Init ; voltmeter, scope on PD3
    CPSIE  I    ; TExaS voltmeter, scope runs on interrupts
    AND R0, #0x0000;
        ADD R0, R0, #16;
        LDR R1, =CYCLE;
        STR R0, [R1];code to store 20% to duty cycle before running loop
loop
    LDR R0,=Blink; Check for breathing
        AND R1, #0x0000;
```

```
        STR R1, [R0];
        LDR R0,=COUNT;
        AND R1, #0x0000;
        STR R1, [R0];
    LDR R0, =GPIO_PORTF_DATA_R;
        LDRSB R1, [R0];
        CMP R1, #0x10;
        BNE AIR;
DUTY  LDR R0, =GPIO_PORTE_DATA_R; Check if external switch is pressed.
        LDRSB R1, [R0];
        CMP R1, #0x02;
        BEQ HIT;
        CMP R1, #0x03;
        BEQ HIT;
ROUTE LDR R0, =CYCLE;Delay subroutine
    LDRSB R1, [R0];
        CMP R1, #0;
        BEQ AGAIN;
DELAY LDR R2, =TERM;
        LDR R0, =GPIO_PORTE_DATA_R;Turn on LED
        LDR R3, [R0];
        ORR R3, #0X01;
        STR R3, [R0];
L1    SUBS R2, #1; Millisecond load
    CMP R2, #0;
        BNE L1;
        SUBS R1, #8;
    CMP R1, #0; on for duty cycle
        BNE DELAY;
```

```
        LDR R0, =CYCLE;
        LDR R1, [R0];
        RSBS R1, R1, #80;
        BEQ loop;
AGAIN LDR R2, =TERM;
        LDR R0, =GPIO_PORTE_DATA_R; Turn off LED
        LDR R3, [R0];
        AND R3, R3, #0x0000;
        STR R3, [R0];
L2      SUBS R2, #1; off for (duty cycle)!
    CMP R2, #0;
        BNE L2;
        CMP R1, #0;
        BEQ loop;
        SUBS R1, #8;
        CMP R1, #0;
        BNE AGAIN;
    B   loop;


HIT     LDR R0, =GPIO_PORTE_DATA_R;check for switch release
        LDRSB R1, [R0];
        CMP R1, #0x02;
        BEQ HIT;
        CMP R1, #0x03;
        BEQ HIT;
        LDR R0, =CYCLE;Duty Cycle subroutine
        LDR R1, [R0];
        ADDS R1, #16; Duty cycle to next level
```

```
        CMP R1, #80; 0X60
        BNE Valid;
        STR R1, [R0];
        SUBS R1, #80;
        B ROUTE;
Valid CMP R1, #96;
        BEQ RESET;
    STR R1, [R0];
        B ROUTE; GOES TO DELAY SUBROUTINE
RESET SUBS R1, #96;
    STR R1, [R0];
        B ROUTE;

        ;Incomplete Breathing LED Code

AIR     LDR R0, =GPIO_PORTF_DATA_R; This makes it blink
    AND R1, #0x0000;
        AND R2, #0x0000;
    LDRSB R1, [R0];
    CMP R1,#0x10;
        BEQ DUTY;
UP      LDR R0, =Blink; Code to increase brightness
        ADDS R8, R8, #4;
        BL MINI;
        CMP R8, #80;
        BEQ Down;
        B UP;
Down  STR R8, [R0];
    LDR R0, =Blink; Code to decrease brightness
```

```
        SUBS R8, #4;
        BL MINI;
        CMP R8, #0;
        BEQ AIR;
        B Down;
MINI  STR R8, [R0];
    LDR R0, =GPIO_PORTF_DATA_R; This makes it blink
    AND R1, #0x0000;
        AND R2, #0x0000;
    LDRSB R1, [R0];
    CMP R1,#0x10;
        BEQ DUTY;
        LDR R0, =COUNT;
        ADD R8, R8, #100;
        LSL R8, R8, #1;
    STR R8, [R0];
    LDR R0, =Blink;Delay subroutine
    LDR R4, [R0]; R4 CONTAINS DUTY CYCLE
    LDR R2, =COUNT; THIS SHOULD BE LARGE NUMBER
        LDR R5, [R2];
        LDR R0, =GPIO_PORTE_DATA_R; OUTPUT STORAGE
        LDR R3, [R0];
        ORR R3, #0X01;
        STR R3, [R0];
L3   SUBS R5, #1;
    CMP R5, #0; Does this line matter?
        BNE L3;
        SUBS R4, #4; PERCENTAGE OF DUTY CYCLE. RUNS X NUMBER OF TIMES
    CMP R4, #0;
```

```
        BNE MINI;
        LDR R0, =Blink;
        LDR R4, [R0];
        RSBS R4, R4, #80;
        ;BEQ Down;
A2  LDR R2, =COUNT;
    LDR R8, [R2];
        LDR R0, =GPIO_PORTE_DATA_R;
        LDR R3, [R0];
        AND R3, R3, #0x0000;
        STR R3, [R0];
L4      SUBS R8, #1;
    CMP R8, #0;
        BNE L4;
        CMP R4, #0;
        ;BEQ AIR;
        SUBS R4, #4;
        ;BNE A2;
    ;B   AIR;
    BX LR;

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