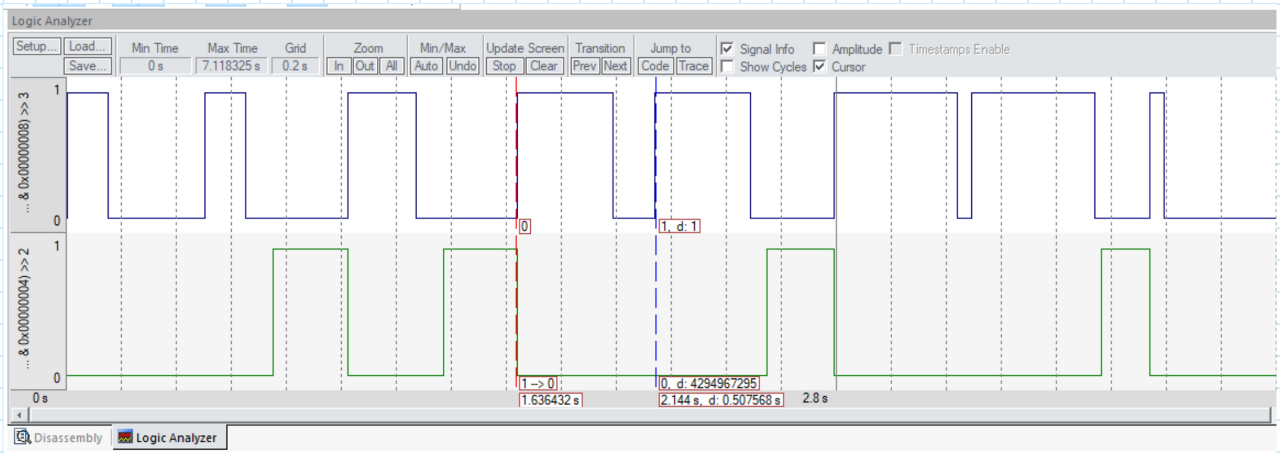
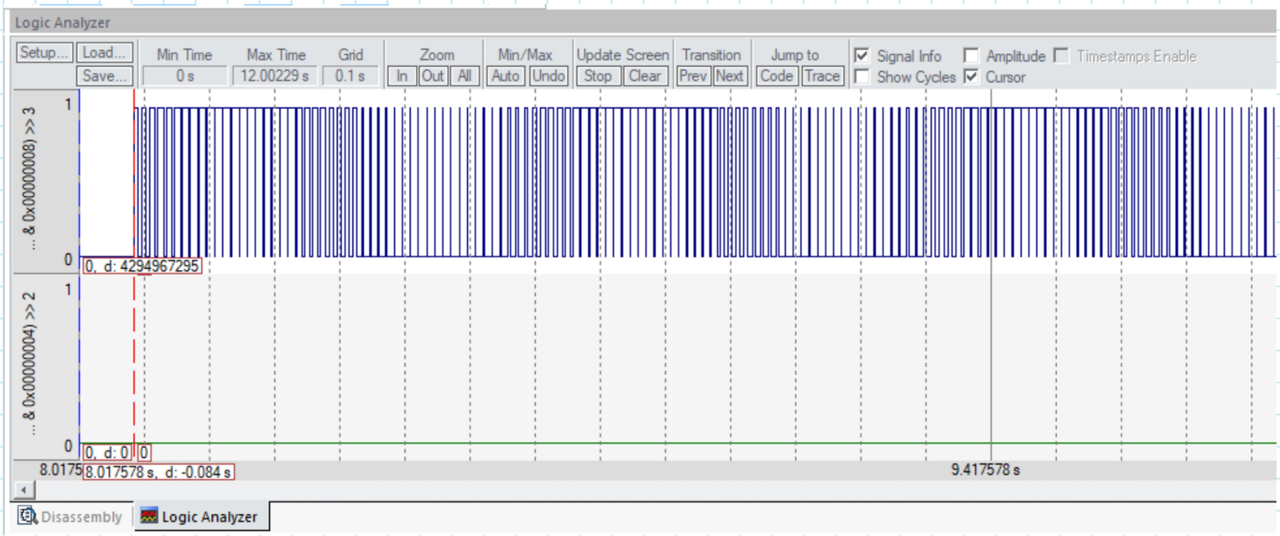
LAB 3

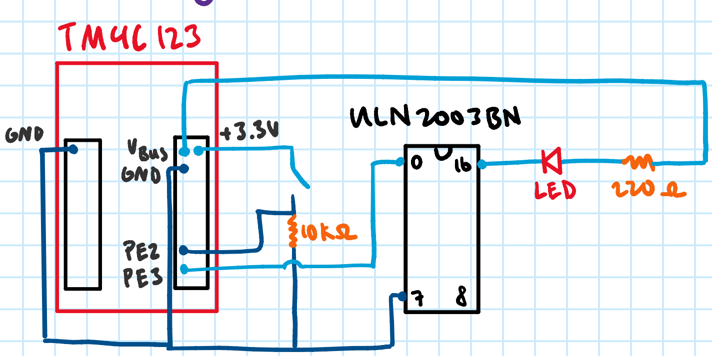
Tuesday, February 19, 2019 8:27 PM

Screenshots





Circuit Diagram



```
;****************** main.s ***************
; Program written by: Zhiyuan Fan and Noah Rose
; Date Created: 2/4/2017
; Last Modified: 1/18/2019
; Brief description of the program
;   The LED toggles at 2 Hz and a varying duty-cycle
; Hardware connections (External: One button and one LED)
;  PE2 is Button input  (1 means pressed, 0 means not pressed)
;  PE3 is LED output (1 activates external LED on protoboard)
;  PF4 is builtin button SW1 on Launchpad (Internal)
;        Negative Logic (0 means pressed, 1 means not pressed)
; Overall functionality of this system is to operate like this
;   1) Make PE3 an output and make PE2 and PF4 inputs.
;   2) The system starts with the the LED toggling at 2Hz,
;      which is 2 times per second with a duty-cycle of 30%.
;      Therefore, the LED is ON for 150ms and off for 350 ms.
;   3) When the button (PE2) is pressed-and-released increase
;      the duty cycle by 20% (modulo 100%). Therefore for each
;      press-and-release the duty cycle changes from 30% to 70% to 70%
;      to 90% to 10% to 30% so on
;   4) Implement a "breathing LED" when SW1 (PF4) on the Launchpad is pressed:
;      a) Be creative and play around with what "breathing" means.
;         An example of "breathing" is most computers power LED in sleep mode
;         (e.g., https://www.youtube.com/watch?v=ZT6siXyIjvQ).
;      b) When (PF4) is released while in breathing mode, resume blinking at 2Hz.
;         The duty cycle can either match the most recent duty-
;         cycle or reset to 30%.
;      TIP: debugging the breathing LED algorithm using the real board.
; PortE device registers
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
GPIO_PORTF_LOCK_R  EQU 0x40025520
GPIO_PORTF_CR_R    EQU 0x40025524
GPIO_LOCK_KEY      EQU 0x4C4F434B  ; Unlocks the GPIO_CR register
SYSCTL_RCGCGPIO_R  EQU 0x400FE608
TWO_HZ_CONST       EQU 0x00061A80
EIGHTY_HZ_CONST    EQU 0x00002710

    IMPORT  TExaS_Init
    THUMB
    AREA |.data|, DATA, READONLY
    ALIGN
SINE_DUTY_ARR DCD 50, 56, 62, 68, 74, 79, 84, 88, 92, 95, 97, 99, 99, 99, 99, 97, 95, 92, 88, 84, 79, 74, 68, 62, 56, 49, 43, 37, 31, 25, 20, 15, 11, 7, 4, 2, 1, 1, 1, 1, 2, 4, 7, 11, 15, 20, 25, 31, 37, 43
    AREA |.text|, CODE, READONLY, ALIGN=2
    THUMB
    EXPORT  Start
```

```
Start
; TExaS_Init sets bus clock at 80 MHz
    BL  TExaS_Init ; voltmeter, scope on PD3
; Initialization goes here
    BL initPortE
    BL initPortF
    CPSIE  I        ; TExaS voltmeter, scope runs on interrupts
; initialize constants
    LDR R0, =GPIO_PORTE_DATA_R
    LDR R7, =GPIO_PORTF_DATA_R
    LDR R8, =SINE_DUTY_ARR
    MOV R11, #0
    LDR R9, =EIGHTY_HZ_CONST
    LDR R1, =TWO_HZ_CONST
    MOV R2, #100    ; for subtracting duty cycle from
    MOV R4, #30     ; inital duty cycle
    MOV R5, #0      ; vector of bools
loop
; main engine goes here
; R3 and R6 used as tmps for calculation

; execute this part if SW1 not pressed
    LDR R6, [R7]
    ANDS R3, R6, #0x10
    BEQ breathe
; check if PE2 high
    LDR R6, [R0]
    ANDS R3, R6, #0x04
    BEQ skip
; poll PE2 until not pressed
poll
    LDR R6, [R0]
    ANDS R3, R6, #0x04
    BNE poll
    ADD R4, #20
    CMP R4, #100
    BLT noreset
    MOV R4, #10
noreset
skip
    BL duty_cycle
breathe

; execute this part if SW1 pressed
    LDR R6, [R7]
    ANDS R3, R6, #0x10
    BNE nobreathe
    LDR R10, [R8, R11]
    BL sin_duty_cycle
    ADD R11, #4
    CMP R11, #196
    BNE iterate
    MOV R11, #0
iterate
nobreathe
    B    loop
```

```

duty_cycle
; R4 has duty cycle out of 100
; toggle E3
    LDR R3, [R0]
    EOR R3, #0x08
    STR R3, [R0]
; delay duty cycle
    MUL R3, R1, R4
delay1
    SUBS R3, #4
    BNE delay1
; toggle E3
    LDR R3, [R0]
    EOR R3, #0x08
    STR R3, [R0]
; delay remainder of period
    SUB R3, R2, R4
    MUL R3, R1
delay2
    SUBS R3, #4
    BNE delay2
    BX LR

sin_duty_cycle
; R10 has duty cycle out of 100; toggle E3
    LDR R3, [R0]
    EOR R3, #0x08
    STR R3, [R0]
; delay duty cycle
    MUL R3, R9, R10
delay3
    SUBS R3, #4
    BNE delay3
; toggle E3
    LDR R3, [R0]
    EOR R3, #0x08
    STR R3, [R0]
; delay remainder of period
    SUB R3, R2, R10
    MUL R3, R9
delay4
    SUBS R3, #4
    BNE delay4
    BX LR

initPortE
; activate clock for port E
    LDR R1, =SYSCTL_RCGCGPIO_R
    LDR R0, [R1]
    ORR R0, #0x10
    STR R0, [R1]
    NOP
    NOP
; E2 input, E3 output
    LDR R1, =GPIO_PORTE_DIR_R
```

```
    LDR R0, [R1]
    ORR R0, #0x08
    STR R0, [R1]
; enable digital IO for E2 and 3
    LDR R1, =GPIO_PORTE_DEN_R
    LDR R0, [R1]
    ORR R0, #0x0C
    STR R0, [R1]
    BX LR

initPortF
; activate clock for port F
    LDR R1, =SYSCTL_RCGCGPIO_R
    LDR R0, [R1]
    ORR R0, #0x20
    STR R0, [R1]
    NOP
    NOP
; unlock port F
    LDR R1, =GPIO_PORTF_LOCK_R
    LDR R0, =GPIO_LOCK_KEY
    STR R0, [R1]
; enable changing F0-F4
    LDR R1, =GPIO_PORTF_CR_R
    LDR R0, [R1]
    ORR R0, #0x1F
    STR R0, [R1]
; F4 is SW1, F0 is SW2, F1-3 is RGB
    LDR R1, =GPIO_PORTF_DIR_R
    LDR R0, [R1]
    ORR R0, #0x0E
    STR R0, [R1]
; enable pull-up resistors for F0 and F4
    LDR R1, =GPIO_PORTF_PUR_R
    LDR R0, [R1]
    ORR R0, #0x11
    STR R0, [R1]
; enable digital IO for F0-4
    LDR R1, =GPIO_PORTF_DEN_R
    LDR R0, [R1]
    ORR R0, #0x1F
    STR R0, [R1]
    BX LR

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

Parameter	Value	Units	Conditions
Resistance of the 10kΩ resistor, R1	9.86 kΩ	ohms	with power off and disconnected from circuit (measured with ohmmeter)
Supply Voltage, $V_{+3.3}$	3 303V	volts	Powered (measured with voltmeter)
Input Voltage, V_{PE2}	0.00V	volts	Powered, but with switch not pressed (measured with voltmeter)
Resistor current	0.00mA	mA	Powered, but switch not pressed $I=V_{PE2}/R1$ (calculated and measured with an ammeter)
Input Voltage, V_{PE2}	3.303V	volts	Powered and with switch pressed (measured with voltmeter)
Resistor current	0.335mA	mA	Powered and switch pressed $I=V_{PE2}/R1$ (calculated and measured with an ammeter)

Table 3.1. Switch measurements.

Figure 3.9. Left: a side view of an LED with leads labeled; Right: the corresponding circuit diagram

Take the measurements as described in Table 3.2. The R19 measurement occurs before R19 is inserted into the circuit. Single step your software to make **PE3** to output. Initially **PE3** will be low. So take four measurements with **PE3** low, rows 2,3,4,5 in Table 3.2. Then, single step some more until **PE3** is high and measure the three voltages (rows 8,9,10 in Table 3.2). When active, the voltage across the LED should be about 2 V, and the LED current should be about 10 mA. The remaining rows are calculated values, based on these 8 measurements. The LED current (row 12) can be determined by calculation or by direct measurement using the ammeter function. You should perform both ways to get LED current.

Warning: NEVER INSERT/REMOVE WIRES/CHIPS WHEN THE POWER IS ON.

Row	Parameter	Value	Units	Conditions
1	Resistance of the 220Ω resistor, R19	219.0Ω	ohms	with power off and disconnected from circuit (measured with ohmmeter)
2	+5 V power supply V_{+5}	4.93V	volts	(measured with voltmeter relative to ground, *notice that the +5V power is not exactly +5 volts*)
3	TM4C123 Output, V_{PE3} input to ULN2003B	0.9mV	volts	with **PE3** = 0 (measured with voltmeter relative to ground). We call this V_{OL} of the TM4C123.
4	ULN2003B Output, pin 16, V_{k-} LED k-	3.633V	volts	with **PE3** = 0 (measured with voltmeter relative to ground). This measurement will be weird, because it is floating.
5	LED a+, V_{a+} Bottom side of R19 (anode side of LED)	4.95V	volts	with **PE3** = 0 (measured with voltmeter relative to ground). This measurement is also weird, because it too is floating.
6	LED voltage	1.297V	volts	calculated as V_{a+} - V_{k-}
7	LED current (off)	5 92mA 5 80 mA	mA	calculated as (V_{+5} - V_{a+})/R19 and measured with an ammeter
8	TM4C123 Output, V_{PE3} input to ULN2003B	3.23 V	volts	with **PE3** = 1 (measured with voltmeter relative to ground). We call this V_{OH} of the TM4C123.
9	ULN2003B Output pin 16, V_{k-} LED k-	0.719V	volts	with **PE3** = 1 (measured with voltmeter relative to ground). We call this V_{OL} or $V_{CE(sat)}$ of the ULN2003B.
10	LED a+, V_{a+} Bottom side of R19 (anode side of LED)	2.636V	volts	with **PE3** = 1 (measured with voltmeter relative to ground)
11	LED voltage	1 917	volts	calculated as V_{a+} - V_{k-}
12	LED current (on)	8.75mA 8.69mA	mA	calculated as (V_{+5} - V_{a+})/R19 and measured with an ammeter

Table 3.2. LED measurements (assuming the 220 Ω resistor is labeled R19 in FIgure 3.8).