Row	Parameter	Value	Units	Conditions
1	Resistance of the 220Ω resistor, R19	216	ohms	with power off and disconnected from circuit (measured with ohmmeter)
2	+5 V power supply V_{+5}	5.12	volts	(measured with voltmeter relative to ground, *notice that the +5V power is not exactly +5 volts*)
3	TM4C123 Output, V_{PE2} input to ULN2003B	.018	volts	with **PE2** = 0 (measured with voltmeter relative to ground). We call this V_{OL} of the TM4C123.
4	ULN2003B Output, pin 16, V_{k-} LED k-	3.82	volts	with **PE2** = 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)	5.1	volts	with **PE2** = 0 (measured with voltmeter relative to ground). This measurement is also weird, because it too is floating.
6	LED voltage	1.28	volts	calculated as V_{a+} - V_{k-}
7	LED current (off)	0.09	mA	calculated as (V_{+5} - V_{a+})/R19 and measured with an ammeter
8	TM4C123 Output, V_{PE2} input to ULN2003B	3.18	volts	with **PE2** = 1 (measured with voltmeter relative to ground). We call this V_{OH} of the TM4C123.
9	ULN2003B Output pin 16, V_{k-} LED k-	0.8	volts	with **PE2** = 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)	2.6	volts	with **PE2** = 1 (measured with voltmeter relative to ground)
11	LED voltage	1.8	volts	calculated as V_{a+} - V_{k-}
12	LED current (on)	11.45	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).

Parameter	Value	Units	Conditions
Resistance of the 10kΩ resistor, R1	986	ohms	with power off and disconnected from circuit (measured with ohmmeter)
Supply Voltage, $V_{+3.3}$	3.28	volts	Powered (measured with voltmeter)
Input Voltage, V_{PE1}	0.00	volts	Powered, but with switch not pressed (measured with voltmeter)
Resistor current	0	mA	Powered, but switch not pressed $I=V_{PE1}/R1$ (calculated and measured with an ammeter)
Input Voltage, V_{PE1}	3.32	volts	Powered and with switch pressed (measured with voltmeter)
Resistor current	3.37	mA	Powered and switch pressed $I=V_{PE1}/R1$ (calculated and measured with an ammeter)

Table 3.1. Switch measurements.

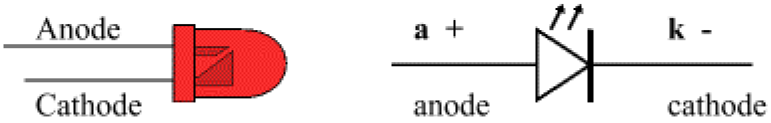
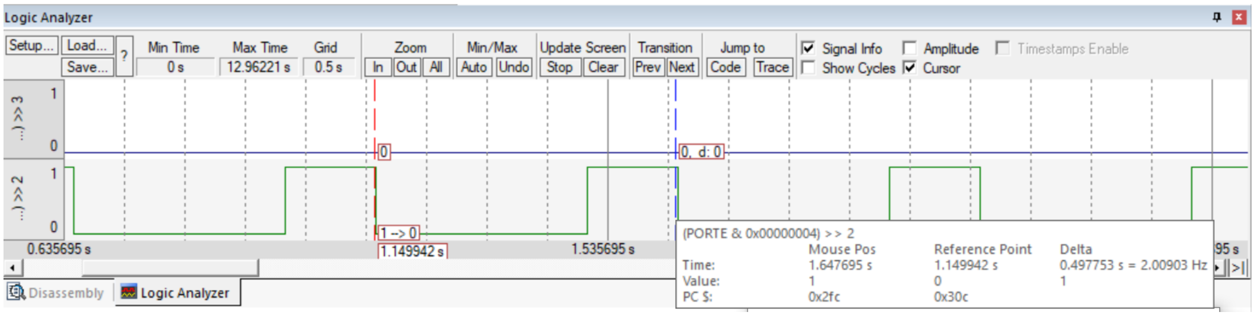


Figure 3.9. Left: a side view of an LED with leads labeled; Right: the corresponding circuit diagram

(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

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

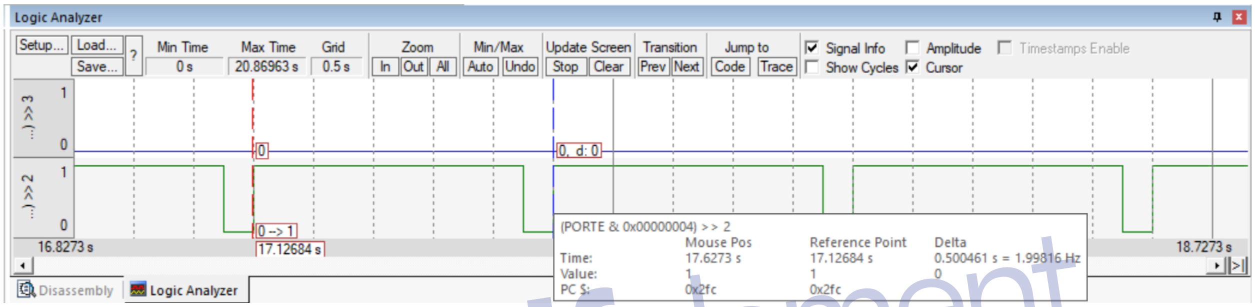
Screenshot 1



Min Time: 0 s
Max Time: 12.96221 s
Grid: 0.5 s
☑ Signal Info ☐ Amplitude ☐ Timestamps Enable
☐ Show Cycles ☑ Cursor

```
(PORTE & 0x00000004) >> 2
          Mouse Pos      Reference Point   Delta
Time:     1.647695 s     1.149942 s        0.497753 s = 2.00903 Hz
Value:    1              0                 1
PC $:     0x2fc          0x30c
```

Screenshot 2



Min Time: 0 s
Max Time: 20.86963 s
Grid: 0.5 s
☑ Signal Info ☐ Amplitude ☐ Timestamps Enable
☐ Show Cycles ☑ Cursor

```
(PORTE & 0x00000004) >> 2
          Mouse Pos      Reference Point   Delta
Time:     17.6273 s      17.12684 s        0.500461 s = 1.99816 Hz
Value:    1              1                 0
PC $:     0x2fc          0x2fc
```

Code:

```
;****************** main.s ***************
; Program written by: Adeel and Zach
; Date Created: 2/4/2017
; Last Modified: 1/20/2020
; Brief description of the program
;   The LED toggles at 2 Hz and a varying duty-cycle
; Hardware connections (External: One button and one LED)
;  PE1 is Button input  (1 means pressed, 0 means not pressed)
;  PE2 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 PE2 an output and make PE1 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 (PE1) 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
```

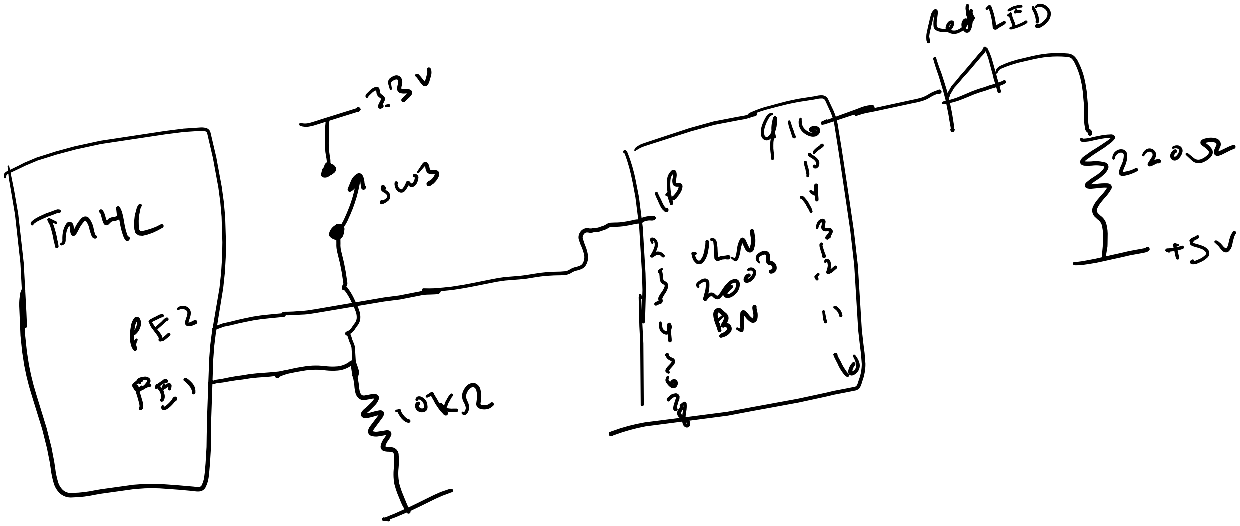
```
GPIO_LOCK_KEY      EQU 0x4C4F434B  ; Unlocks the GPIO_CR register

SYSCTL_RCGCGPIO_R  EQU 0x400FE608


    IMPORT  TExaS_Init

    THUMB

    AREA    DATA, ALIGN=2

;global variables go here



    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

   LDR R0, =SYSCTL_RCGCGPIO_R

         LDRB R1, [R0]

         ORR R1, #0x30

         STRB R1, [R0]

         NOP

         NOP


         LDR R0, =GPIO_PORTF_LOCK_R

         LDR R1, =GPIO_LOCK_KEY

         STR R1, [R0]

         LDR R0, =GPIO_PORTF_CR_R

         LDR R1, [R0]

         ORR R1, #0xFF
         STR R1, [R0]

         LDR R0, =GPIO_PORTF_DIR_R
         LDR R1, [R0]
         AND R1, #0xFF
         STR R1, [R0]

         LDR R0, =GPIO_PORTF_DEN_R
         LDR R1, [R0]
         ORR R1, #0x10
         STR R1, [R0]

         LDR R0, =GPIO_PORTF_PUR_R
         LDR R1, [R0]
         ORR R1, #0x10
         STR R1, [R0]

         LDR R0, =GPIO_PORTE_DIR_R
         LDR R1, [R0]
         AND R1, #0xFD
         ORR R1, #0x04
         STR R1, [R0]

         LDR R0, =GPIO_PORTE_DEN_R
         LDR R1, [R0]
         ORR R1, #0x06
         STR R1, [R0]

   CPSIE  I    ; TExaS voltmeter, scope runs on interrupts
```

```
        MOV R3, #50 ;CENTISECODNS CONSTANT
        MOV R2, #3  ;pERCENT COUNTER
        MOV R4, #0  ;PREVIOUS VALUE
        MOV R7, #10 ;MODULO VALUE
loop
; main engine goes here
        LDR R0, =GPIO_PORTE_DATA_R
        LDR R8, [R0]
       ; AND R1, R8, #0x02
        ;LSR R1, R1, #1
        ;MOV R9, R1
        ;EOR R1, R1, R4
        ;AND R1, R1, R4
        ;MOV R4, R9
        ;ADDS R1, R1, #0
        ;BNE INCREMENT
        MUL R5, R2, R3
        ORR R8, R8, #0x04
        STR R8, [R0]
        BL delay
        LDR R8, [R0]
        AND R8, R8, #0xFB
        STR R8, [R0]
        ADD R5, R2, #-10
        EOR R5, #0xFFFFFFFF
        ADD R5, R5, #1
        MUL R5, R3, R5
        BL delay
```

```
            AND R1, R8, #0x02
            LSR R1, R1, #1
            MOV R9, R1
            EOR R1, R1, R4
            AND R1, R1, R4
            MOV R4, R9
            ADDS R1, R1, #0
            BNE INCREMENT

    LDR R12, =GPIO_PORTF_DATA_R
    LDR R12, [R12]
    ;LSR R0, #4
    AND R12, #0x10
    ADDS R12, R12, #0

    BEQ Breathe

    B    loop
INCREMENT
            ADD R2, R2, #2
            UDIV R6, R2, R7
            MLS R2, R6, R7, R2
            BX LR



delay ;Delays for parameter 1 ms

  PUSH {R1, R5}
  MOV R1, #5333
  MUL R5, R5, R1
wait
  SUBS R5, R5, #0x01
  LDR R8, [R0]
  AND R1, R8, #0x02
  LSR R1, R1, #1
  MOV R9, R1
  EOR R1, R1, R4
  AND R1, R1, R4
  MOV R4, R9
  ADDS R1, R1, #0

  BNE INCREMENT

  ADDS R5, R5, #0
  BNE  wait
  POP {R1, R5}

  BX LR

Breathe
  ;R5: address of port
  ;R1: duty cycle
  ;R3: cycle counter
  ;R4: flag for incimenting vs decrementing
  PUSH {R0, R1, R3, R5, R7, R11}
  MOV R5, R2
  MOV R3, #1
  MOV R11, #1
```

```
bLoop

 MOV R1, R5



 PUSH {R14, R12}
 LDR R7, =GPIO_PORTE_DATA_R
 LDR R12, [R7]
 ORR R12, R12, #0x04
 STR R12, [R7]

 BL delayUS
 MOV R5, R1
 AND R12, R12, #0xFB
 STR R12, [R7]
 EOR R5, R1, #0xFFFFFFFF
 ADD R5, R5, #101



 BL delayUS

 MOV R5, R1

 POP {R14, R12}

 SUBS R3, R3, #1
 BNE bLoop
 MOV R3, #1 ;speed
 ADDS R5, R11, R5
```

```
  BEQ ifZero
  SUBS R12, R5, #100
  BEQ ifTen
  B neither
ifZero
  MOV R11, #1
  ADD R5, R5, #1
  B neither
ifTen
  MOV R11, #-1
  SUB R5, R5, #1
neither


  LDR R1, =GPIO_PORTF_DATA_R
  LDR R1, [R1]
  ;LSR R0, #4
  AND R1, #0x10
  ADDS R1, R1, #0

  BEQ bLoop

  POP {R0, R1, R3, R5, R6, R11}
  BX LR

delayUS ;Delays for parameter 1 microseconds

  PUSH {R1, R5}
  MOV R1, #533
  MUL R5, R5, R1

waitUS

  SUBS R5, R5, #0x01

  LDR R8, [R0]

  AND R1, R8, #0x02

  LSR R1, R1, #1

  MOV R9, R1

  EOR R1, R1, R4

  AND R1, R1, R4

  MOV R4, R9

  ADDS R1, R1, #0


  BNE INCREMENT


  ADDS R5, R5, #0

  BNE  waitUS

  POP {R1, R5}


  BX LR



    ALIGN      ; make sure the end of this section is aligned

    END        ; end of file
```


TM4C
PE2
PE1
3.3V
SW3
10kΩ
ULN
2003
BN
1
2
3
4
5
6
7
8
9
16
15
14
13
12
11
10
Red LED
220Ω
+5V