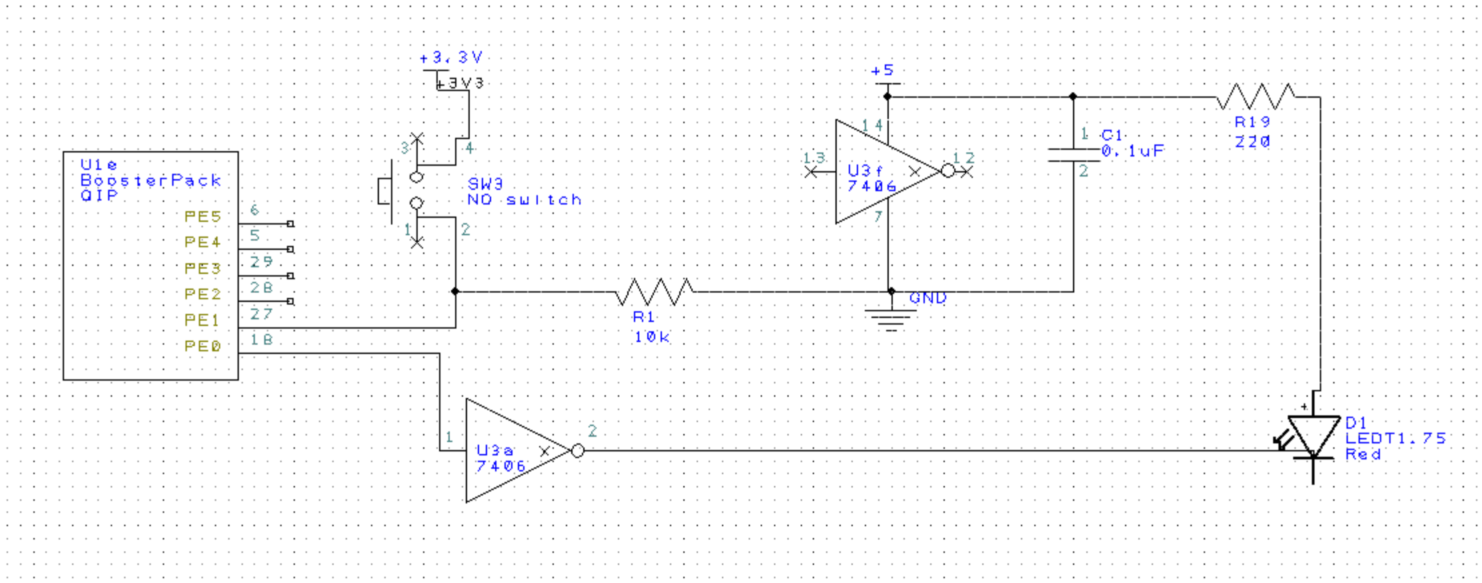
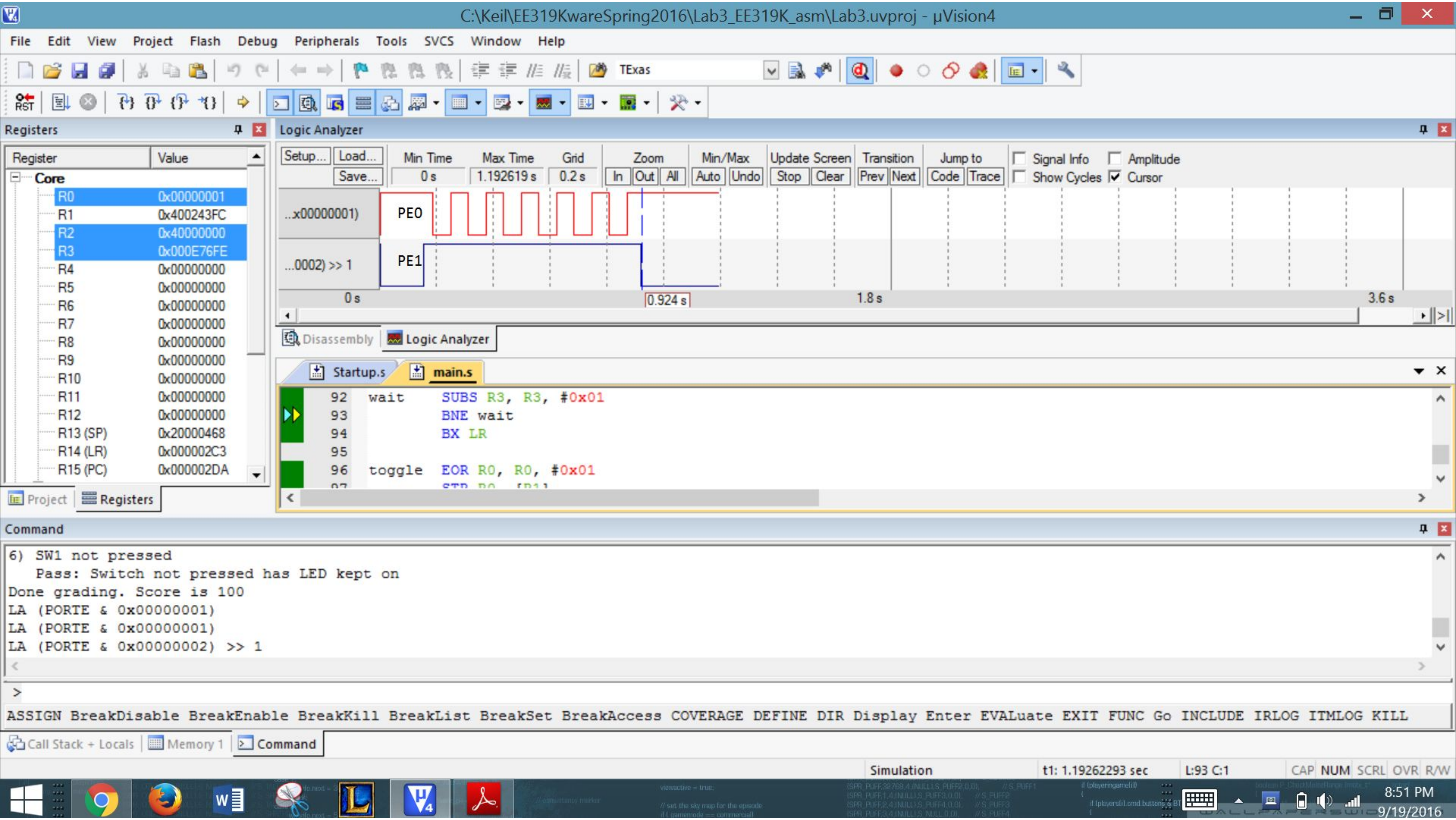

+3.3V
+3V3
3
4
SW3
NO switch
1
2
U1e
BoosterPack
QIP
PE5
6
PE4
5
PE3
29
PE2
28
PE1
27
PE0
18
+5
14
13
U3f
7406
12
7
GND
1
C1
0.1uF
2
R19
220
R1
10k
1
U3a
7406
2
+
D1
LEDT1.75
Red



Min Time: 0 s
Max Time: 1.192619 s
Grid: 0.2 s

Register	Value
Core	
R0	0x00000001
R1	0x400243FC
R2	0x40000000
R3	0x000E76FE
R4	0x00000000
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20000468
R14 (LR)	0x000002C3
R15 (PC)	0x000002DA

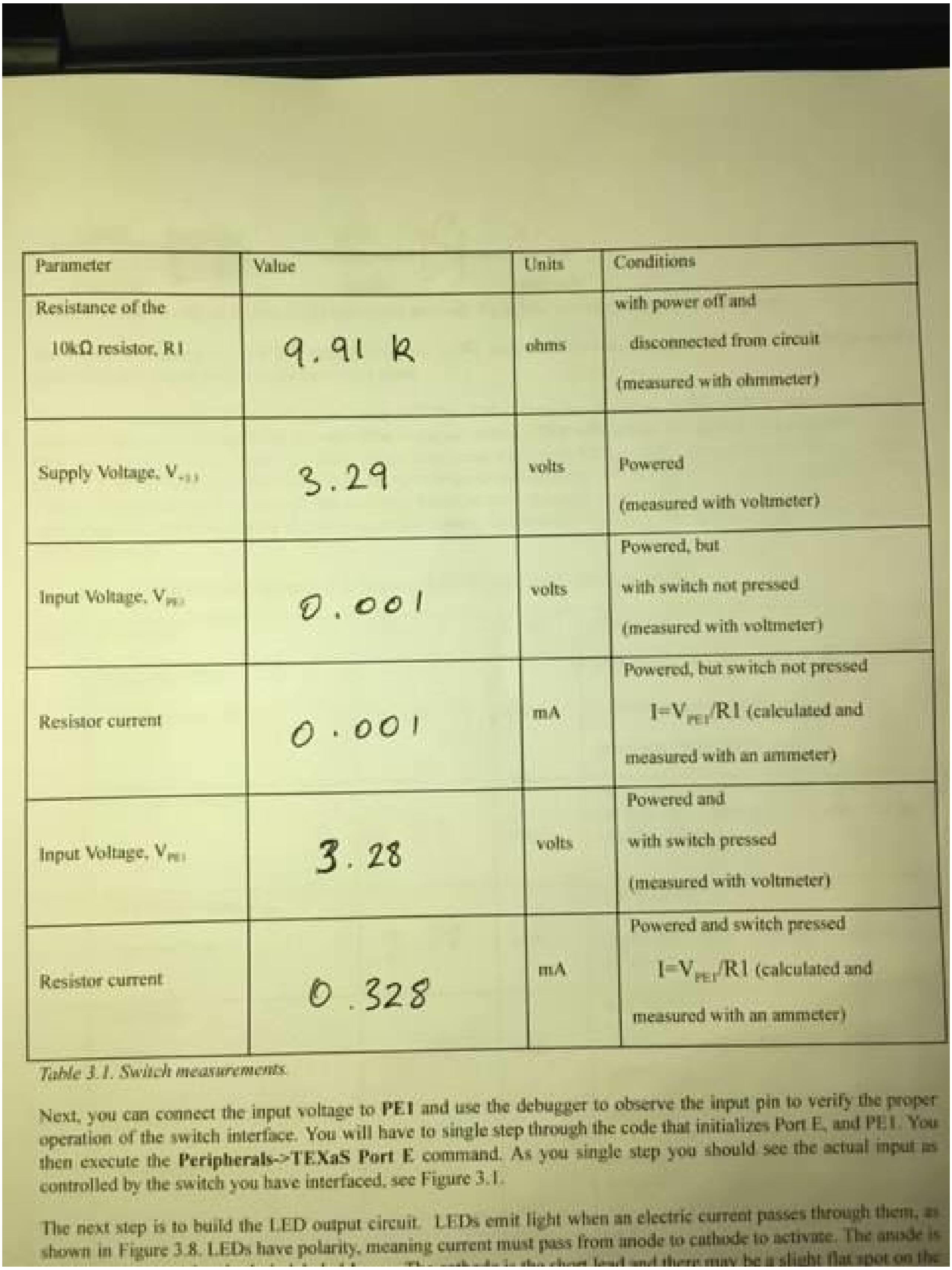
```
wait    SUBS R3, R3, #0x01
        BNE wait
        BX LR

toggle  EOR R0, R0, #0x01
        STR R0, [R1]
```

```
6) SW1 not pressed
   Pass: Switch not pressed has LED kept on
Done grading. Score is 100
LA (PORTE & 0x00000001)
LA (PORTE & 0x00000001)
LA (PORTE & 0x00000002) >> 1
```

```
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR Display Enter EVALuate EXIT FUNC Go INCLUDE IRLOG ITMLOG KILL
```

Simulation   t1: 1.19262293 sec   L:93 C:1   CAP NUM SCRL OVR R/W



Parameter	Value	Units	Conditions
Resistance of the 10kΩ resistor, R1	9.91 K	ohms	with power off and disconnected from circuit (measured with ohmmeter)
Supply Voltage, $V_{+3.3}$	3.29	volts	Powered (measured with voltmeter)
Input Voltage, V_{PE1}	0.001	volts	Powered, but with switch not pressed (measured with voltmeter)
Resistor current	0.001	mA	Powered, but switch not pressed $I=V_{PE1}/R1$ (calculated and measured with an ammeter)
Input Voltage, V_{PE1}	3.28	volts	Powered and with switch pressed (measured with voltmeter)
Resistor current	0.328	mA	Powered and switch pressed $I=V_{PE1}/R1$ (calculated and measured with an ammeter)

Table 3.1. Switch measurements.

Next, you can connect the input voltage to **PE1** and use the debugger to observe the input pin to verify the proper operation of the switch interface. You will have to single step through the code that initializes Port E, and PE1. You then execute the **Peripherals->TEXaS Port E** command. As you single step you should see the actual input as controlled by the switch you have interfaced, see Figure 3.1.

The next step is to build the LED output circuit. LEDs emit light when an electric current passes through them, as shown in Figure 3.8. LEDs have polarity, meaning current must pass from anode to cathode to activate. The anode is [illegible] The cathode is the short lead and there may be a slight flat spot on the

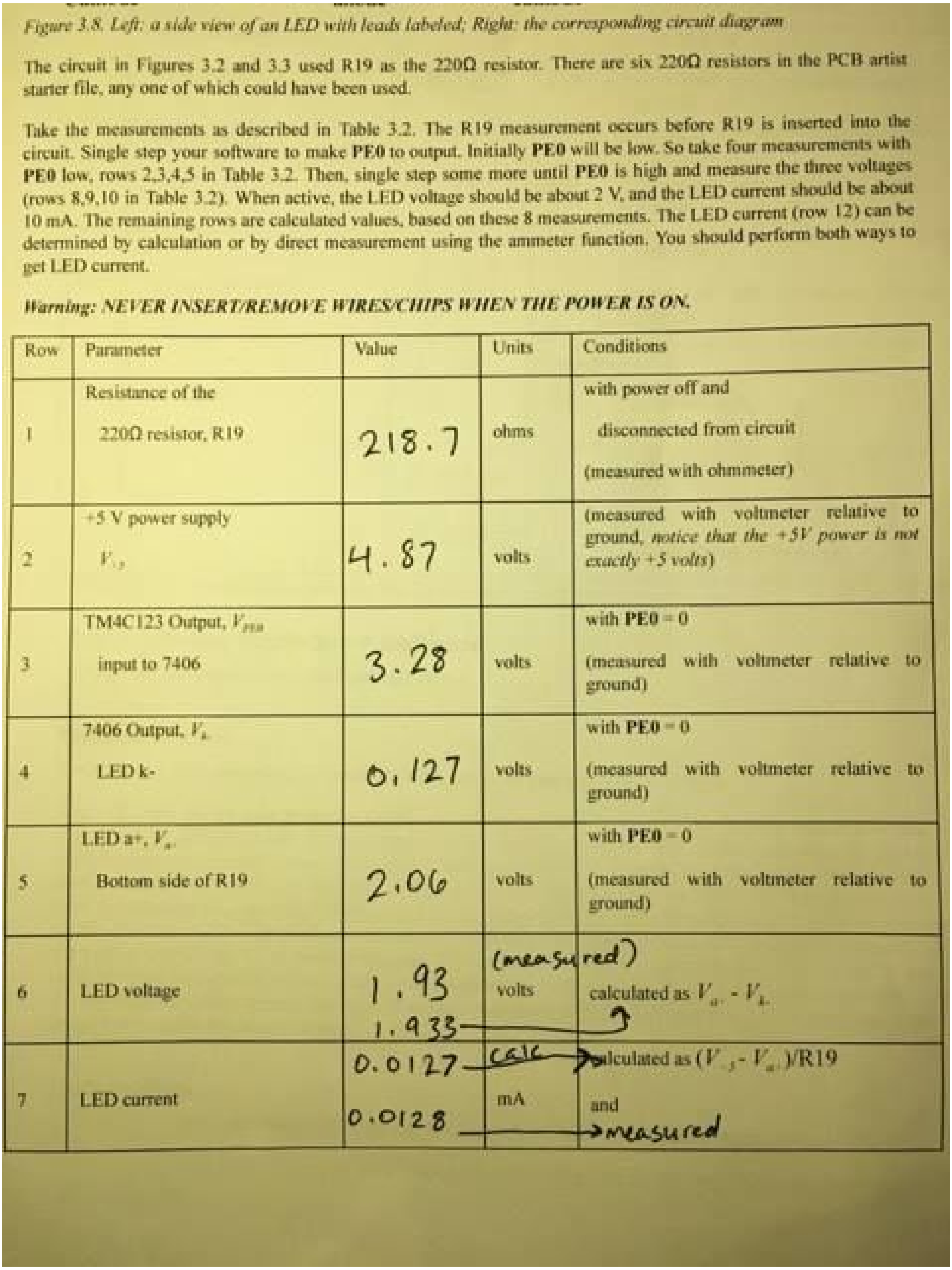


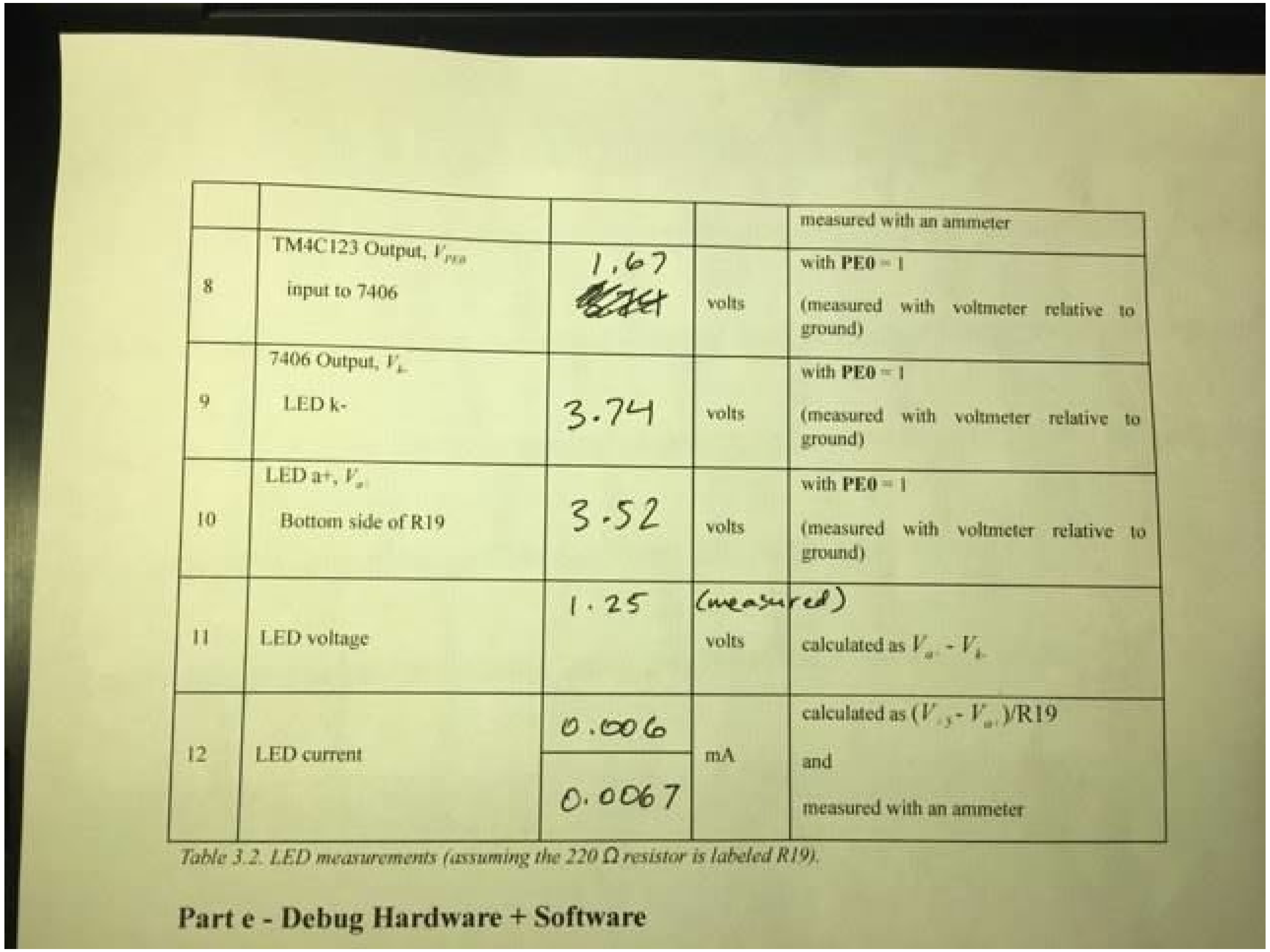
Figure 3.8. Left: a side view of an LED with leads labeled; Right: the corresponding circuit diagram

The circuit in Figures 3.2 and 3.3 used R19 as the 220Ω resistor. There are six 220Ω resistors in the PCB artist starter file, any one of which could have been used.

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 **PE0** to output. Initially **PE0** will be low. So take four measurements with **PE0** low, rows 2,3,4,5 in Table 3.2. Then, single step some more until **PE0** is high and measure the three voltages (rows 8,9,10 in Table 3.2). When active, the LED voltage 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	218.7	ohms	with power off and disconnected from circuit (measured with ohmmeter)
2	+5 V power supply V_{+5}	4.87	volts	(measured with voltmeter relative to ground, *notice that the +5V power is not exactly +5 volts*)
3	TM4C123 Output, V_{PE0} input to 7406	3.28	volts	with **PE0** = 0 (measured with voltmeter relative to ground)
4	7406 Output, V_{k-} LED k-	0.127	volts	with **PE0** = 0 (measured with voltmeter relative to ground)
5	LED a+, V_{a+} Bottom side of R19	2.06	volts	with **PE0** = 0 (measured with voltmeter relative to ground)
6	LED voltage	1.93 1.933	(measured) volts	calculated as $V_{a+} - V_{k-}$
7	LED current	0.0127 0.0128	calc mA	calculated as $(V_{+5} - V_{a+})$/R19 and measured



				measured with an ammeter
8	TM4C123 Output, V_{PE0} input to 7406	1.67	volts	with **PE0** = 1 (measured with voltmeter relative to ground)
9	7406 Output, V_{k-} LED k-	3.74	volts	with **PE0** = 1 (measured with voltmeter relative to ground)
10	LED a+, V_{a+} Bottom side of R19	3.52	volts	with **PE0** = 1 (measured with voltmeter relative to ground)
11	LED voltage	1.25 (measured)	volts	calculated as $V_{a+} - V_{k-}$
12	LED current	0.006 0.0067	mA	calculated as $(V_{3.3} - V_{a+})$/R19 and measured with an ammeter

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

Part e - Debug Hardware + Software

```
;****************** main.s ***************
; Program written by: ***Ankith Kandikonda & Elvin Galarza***
; Date Created: 1/22/2016
; Last Modified: 1/22/2016
; Section ***Tuesday 3-4pm***
; Instructor: ***Vijay Janapa Reddi***
; Lab number: 3
; Brief description of the program
;   If the switch is presses, the LED toggles at 8 Hz
; Hardware connections
;  PE1 is switch input  (1 means pressed, 0 means not pressed)
;  PE0 is LED output (1 activates external LED on protoboard)
;Overall functionality of this system is the similar to Lab 2, with six changes:
;1-  the pin to which we connect the switch is moved to PE1,
;2-  you will have to remove the PUR initialization because pull up is no longer needed.
;3-  the pin to which we connect the LED is moved to PE0,
;4-  the switch is changed from negative to positive logic, and
;5-  you should increase the delay so it flashes about 8 Hz.
;6-  the LED should be on when the switch is not pressed
; Operation
;   1) Make PE0 an output and make PE1 an input.
;   2) The system starts with the LED on (make PE0 =1).
;   3) Wait about 62 ms
;   4) If the switch is pressed (PE1 is 1), then toggle the LED once, else turn the LED on.
;   5) Steps 3 and 4 are repeated over and over


GPIO_PORTE_DATA_R       EQU   0x400243FC
GPIO_PORTE_DIR_R        EQU   0x40024400
```

```
GPIO_PORTE_AFSEL_R     EQU   0x40024420

GPIO_PORTE_DEN_R       EQU   0x4002451C

GPIO_PORTE_AMSEL_R     EQU   0x40024528

GPIO_PORTE_PCTL_R      EQU   0x4002452C

SYSCTL_RCGCGPIO_R      EQU   0x400FE608

TIMER                                     EQU       1240000


    IMPORT  TExaS_Init

    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

; you initialize PE1 PE0


        LDR R1, =SYSCTL_RCGCGPIO_R                        ;turns on clock for PortE

        LDR R0, [R1]

        ORR R0, R0, #0x10

        STR R0, [R1]

        NOP

        NOP


        LDR R1, =GPIO_PORTE_DIR_R                         ;friendly code for making PE0 as output
and PE1 an input

        ORR R0, R0, #0x01

        STR R0, [R1]
        LDR R1, =GPIO_PORTE_DEN_R                   ;indicates we want PE0-1 to be used digitally
        LDR R0, [R1]
        ORR R0, R0, #0x03
        STR R0, [R1]

        LDR R1, =GPIO_PORTE_AFSEL_R                 ;turns off Alternative functions for PE0-1
        LDR R0, [R1]
        AND R0, R0, #0xFC
        STR R0, [R1]

        LDR R1, =GPIO_PORTE_DATA_R                  ;makes PE0 on, originally
        LDR R0, [R1]
        ORR R0, R0, #0x01
        STR R0, [R1]

   CPSIE  I    ; TExaS voltmeter, scope runs on interrupts

loop
; you input output delay
                BL delay
                LDR R1, =GPIO_PORTE_DATA_R
                LDR R0, [R1]
                LSLS R2, R0, #30                    ;puts PE1 in most sig bit
                BMI toggle

                LDR R1, =GPIO_PORTE_DATA_R
                LDR R0, [R1]

                ORR R0, R0, #0x01

                STR R0, [R1]

                B loop



delay   LDR R3, =TIMER

wait    SUBS R3, R3, #0x01

                BNE wait

                BX LR


toggle  EOR R0, R0, #0x01

                STR R0, [R1]


   B    loop





   ALIGN    ; make sure the end of this section is aligned

   END      ; end of file
```