kuntalbahennarshibhaipatel@my.unt.edu
11085707
Zacarith Watkins
Zacarithwatkins@my.unt.edu
10918041

Demonstration:

Demonstrate turning on of the red LED when push button S1 is pressed and turning off of the red LED when the S1 is again pressed _____

Demonstrate turning on of the green LED when push button S2 is pressed and turning off of the green LED when the S2 is again pressed _____

Demonstrate that there is no bouncing of switches when the switches S1 and S2 are pressed. _____

Requirements:

Turn the red LED (RGB LED) on when push button S1 is pressed and turn it off when S1 is again pressed _____

Turn the green LED (RGB LED) on when push button S2 is pressed and turn it off when S2 is again pressed _____

Avoid debouncing of the switches (S1andS2). Do not use any delay functions or delay loops to handle debouncing of the switches and make sure interrupts are not used.

_____

Learning Objectives:
The general learning objectives of this lab were to figure out how to set up our new Arduino board by implementing a debounced switch to turn a LED on and off. Other learning objectives include working with bread boards for those who have never done so.

General Steps:
The general steps needed to complete this lab were

1. Go to lab and put your board together with the breadboard
2. Install LEDs to correct pins
3. Install switches.
4. Download software to work with Arduino board
5. Code the necessary instruction for the Lab
6. Write Report.

kuntalbahennarshibhaipatel@my.unt.edu
11085707
Zacarith Watkins
Zacarithwatkins@my.unt.edu
10918041

Detailed Steps:

Some detailed steps to complete this lab were . . . .

1. Setup the ESP8266 Module on a breadboard and connect the micro USB cable to the USB port of the board. Do not connect to computer
2. Connect the anode of the red LED (RGB LED) to port D0 with a series 1 KΩ resistor and connect the anode of the green LED (RGB LED) to port D1 with a series 1 KΩ resistor. Also, connect the common cathode to ground
3. Connect one pin of the push button (tactile) switch (S1) to D2. Pull the pin D2 to Ground (G) using a 1 KΩ resistor. Connect the other pin of the switch (S1) to 3.3V
4. Connect one pin of the push button (tactile) switch (S2) to D3. Pull the pin D3 to Ground (G) using a 1 KΩ resistor. Connect the other pin of the switch (S2) to 3.3V
5. Install and setup Arduino IDE 1.8.6 from https://www.arduino.cc/en/Main/Software
6. Follow the instructions and install the required board manager from http://henrysbench.capnfatz.com/henrys-bench/arduino-projects-tips-and-more/arduino-esp8266-lolin-nodemcu-getting-started/
7. Run the example code from File→Examples→ESP8266→blink and you should see red LED blinking
8. Arduino IDE reference is available here https://www.arduino.cc/reference/en/
9. Use the example codes and the Arduino reference to meet the requirements and complete the lab
10. Record your observations in the report.

Samples: Most code was derived from sample/example codes on Arduino.

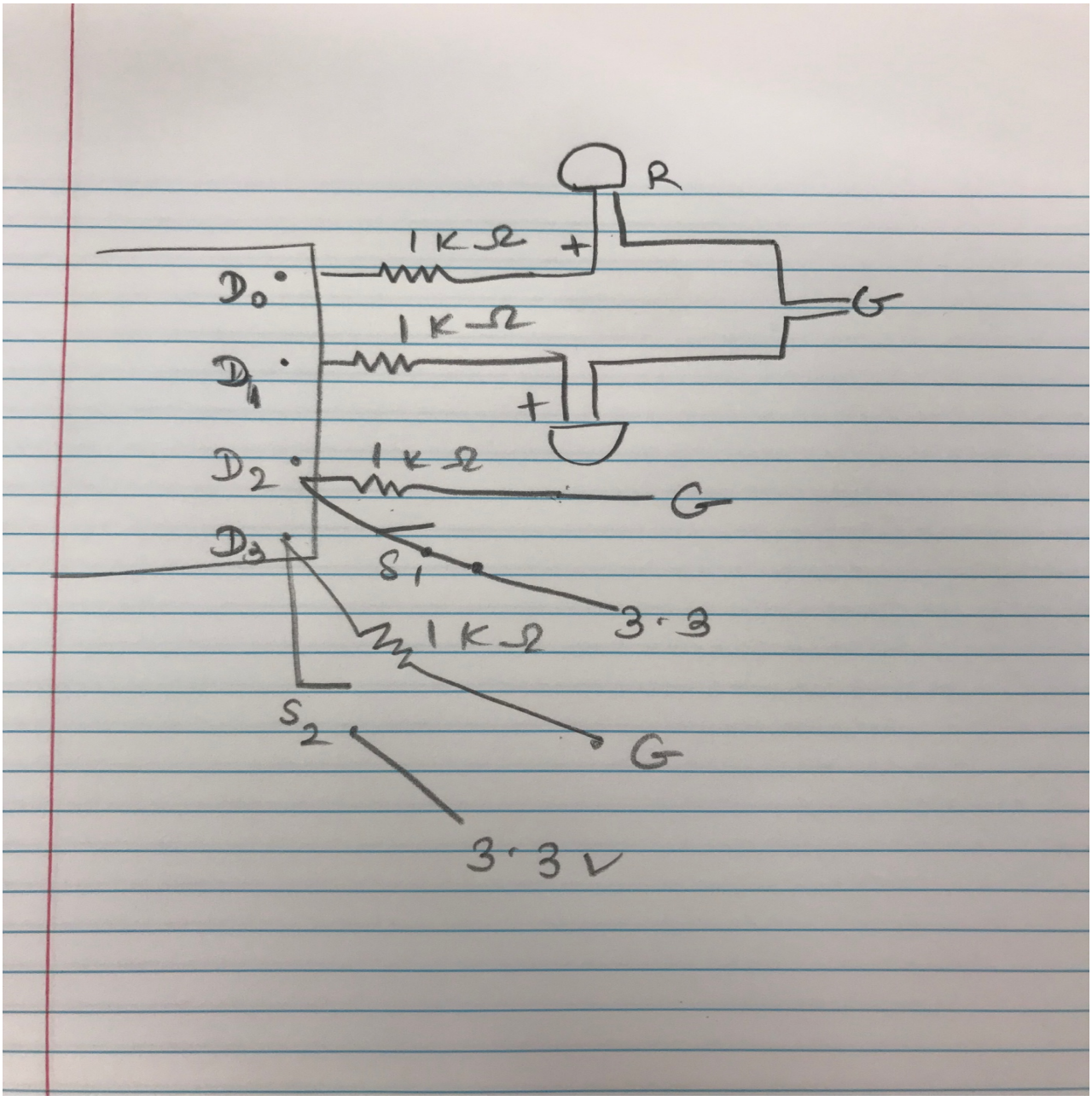
```
void setup() {
  pinMode(Rled, OUTPUT);
  pinMode(Gled, OUTPUT);
  pinMode(s1, INPUT);
  pinMode(s2, INPUT);
}
```

11085707
Zacarith Watkins
Zacarithwatkins@my.unt.edu
10918041

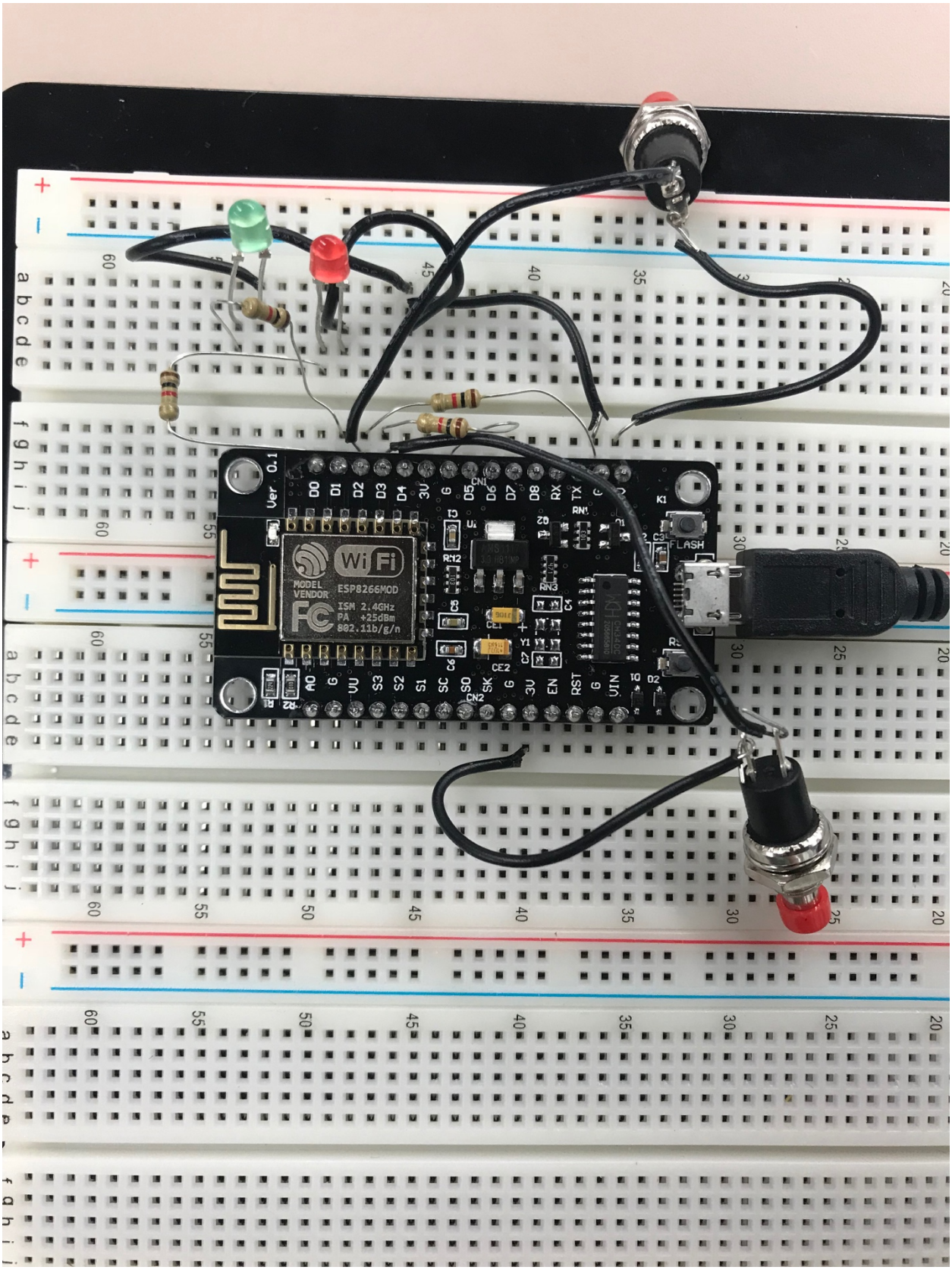
```
void loop() {
//red

  buttonState = digitalRead(s1);

  //filter out any noise by setting a time buffer
  if ( (millis() - lastDebounceTime) > debounceDelay) {
```



kuntalbahennarshibhaipatel@my.unt.edu
11085707
Zacarith Watkins
Zacarithwatkins@my.unt.edu
10918041



kuntalbahennarshibhaipatel@my.unt.edu
11085707
Zacarith Watkins
Zacarithwatkins@my.unt.edu
10918041

Observations:
Some important observations while completing/testing this lab were that it was not so easy to figure out that millis() was an acceptable function for this Lab. Other observations are that sometimes when compiling code, you will receive errors when trying to upload code. (or at least that is what happened for us.) Other than that, this program was very insightful and is going to help us get started for the semester.

Summary:
In this lab we learned how to implement a software debounce without using delay functions/interrupts. We also learned a few bread board implementation tactics along with learning how to implement software that can work with hardware using digital input/outputs. Overall this lab was very insightful and will help set us up for success for the labs to come.