

Roger Burns II

CEIS101 Final Project – IoT Home Security System



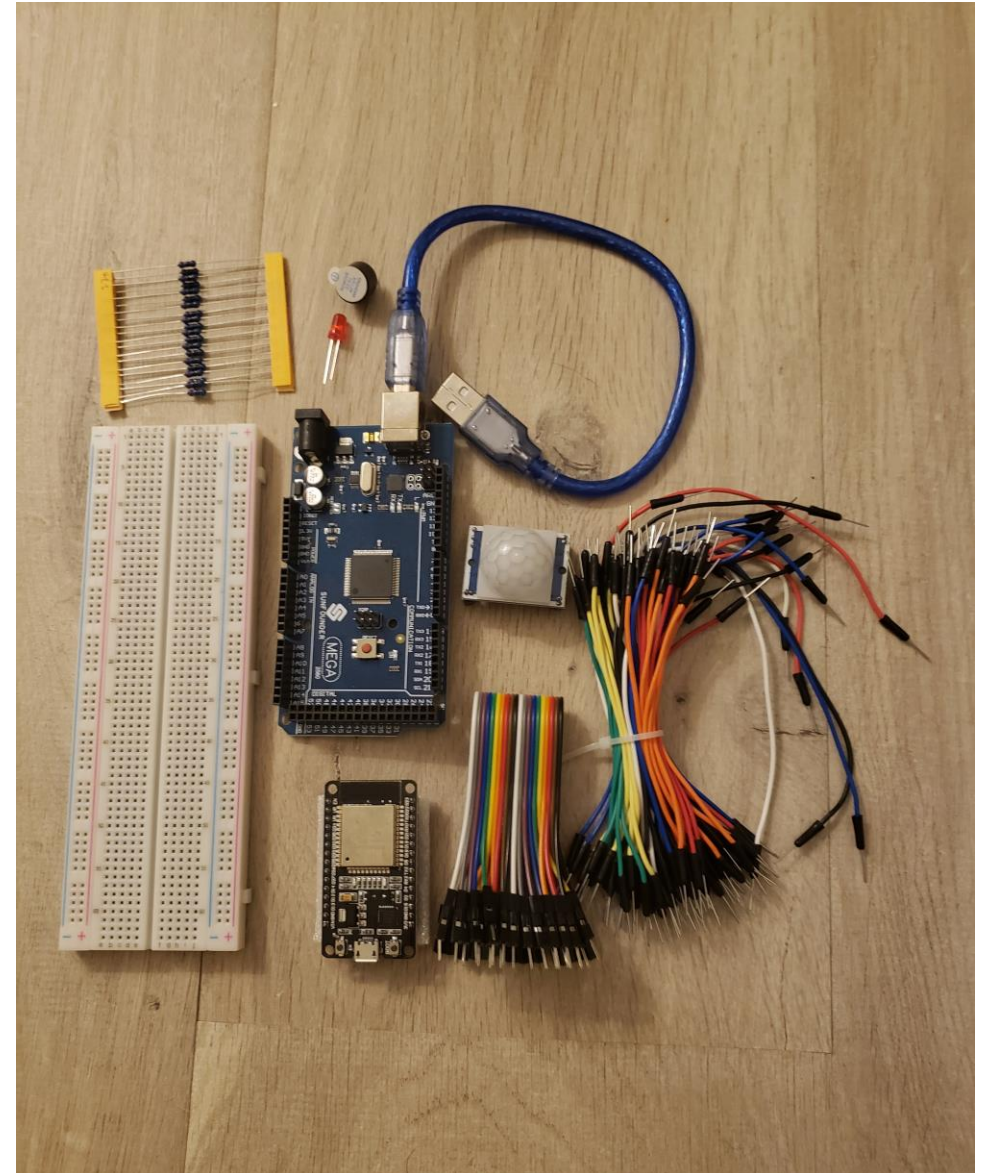
Objective:

With the IoT expanding at an astronomical rate this course brings the basic building blocks to life and develops a home security system.



Inventory and Organization:

- Arduino Megaboard
- Resistor - 220Ω
- LED
- Breadboard
- Motion Sensor
- Wire(s)
- Buzzer
- Esp32 Board



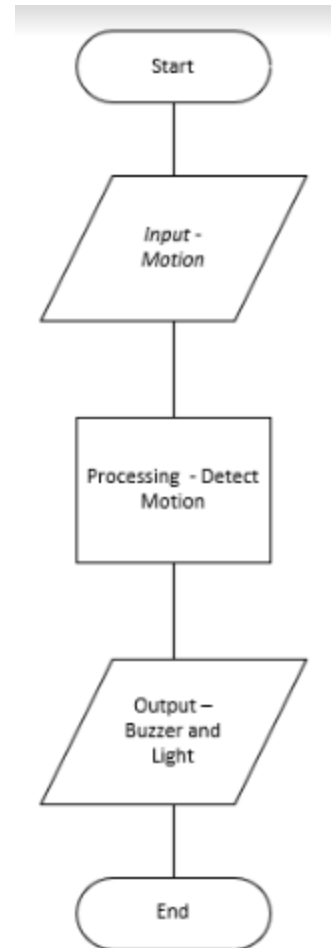


Organizations Using Flowcharts:

- Every company has processes and tasks which require a flowchart, however most commonly used in technical and engineering settings; Google, Intel, etc.
 - This helps cut down on any technical jargon.
 - Helps to streamline processes in turn increasing productivity.
 - Helps to troubleshoot and fix problems.
-

Flowchart

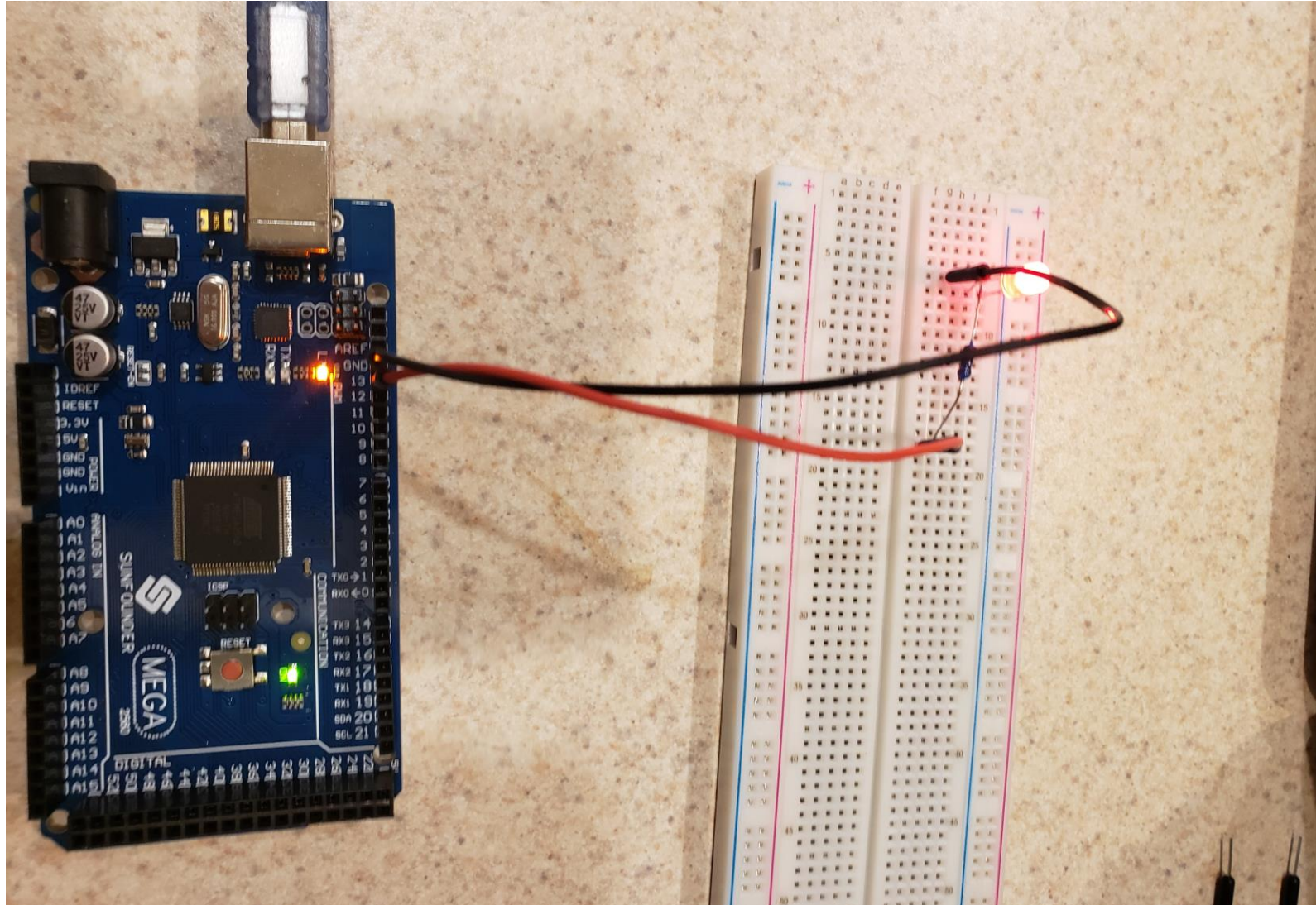
Motion was detected by a sensor,
processed and an audible and visual alert
was sent.





Input – Processing - Output

Input	Process	Output
Motion	Detect Motion	Buzzer and light



LED Light
Circuit

Code in Arduino IDE:

etch_feb02a | Arduino 1.8.5

.dit Sketch Tools Help



etch_feb02a\$

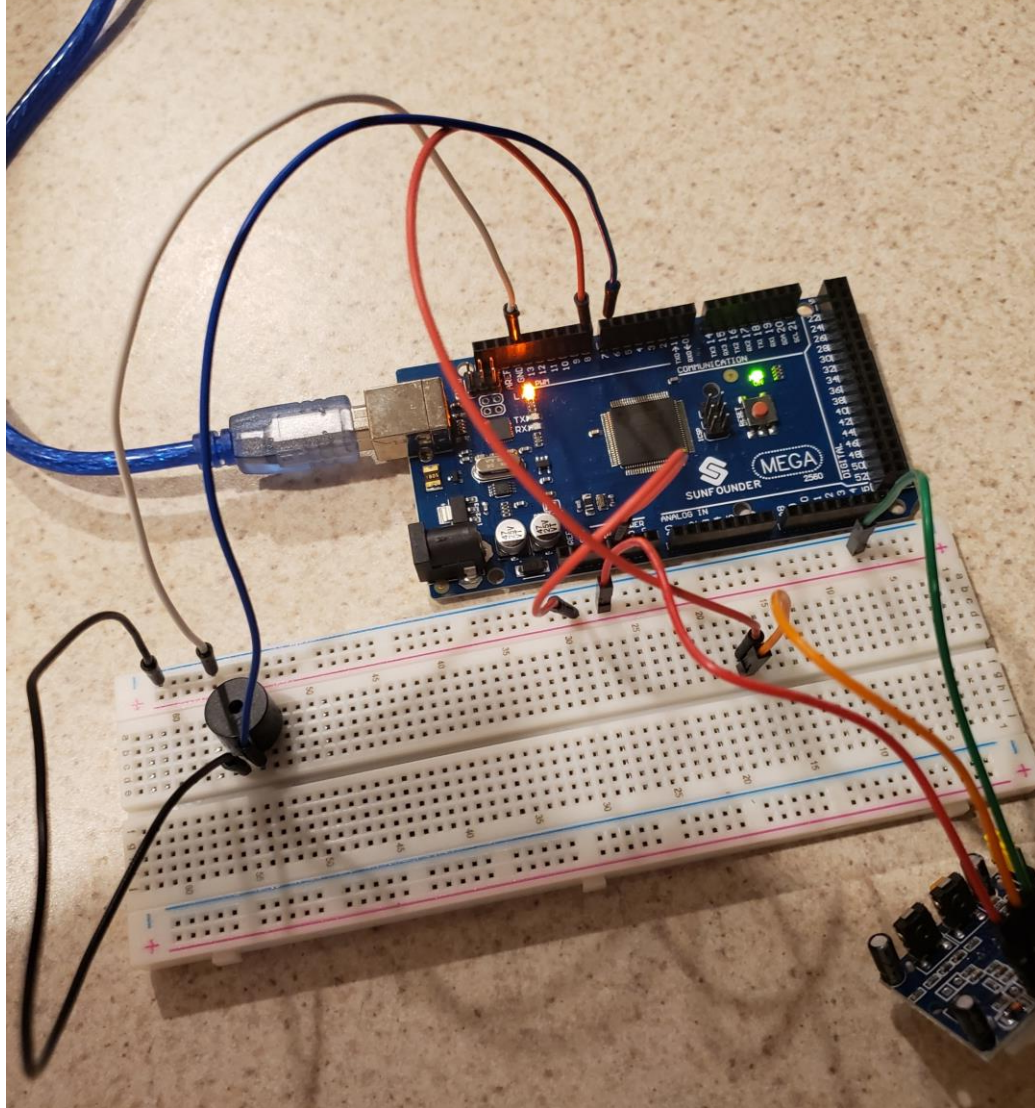
he setup function runs once when you press reset or power the board

```
setup() {  
  pinMode(13, OUTPUT); // configure pin 13 as an output
```


he loop function runs over and over again forever

```
loop() {  
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000);             // wait for a second  
  digitalWrite(13, LOW);  // turn the LED off by making the voltage LOW  
  delay(1000);             // wait for a second
```

Arduino/Genuino Uno on C



Motion Sensor w/Buzzer Circuit



Buzzer code in Arduino IDE:

```
int buzzerPin = 7;           // choose the pin for the buzzer
int inputPin = 8;            // choose the input pin (for PIR sensor)
int pirState = LOW;          // we start, assuming no motion detected
int val = 0;                 // variable for reading the pin status

void setup() {

  pinMode(buzzerPin, OUTPUT); // declare buzzer as output
  pinMode(inputPin, INPUT);   // declare sensor as input

}

void loop(){
  val = digitalRead(inputPin); // read input value
  if (val == HIGH) {           // check if the input is HIGH
    digitalWrite(buzzerPin, HIGH); // turn buzzer ON

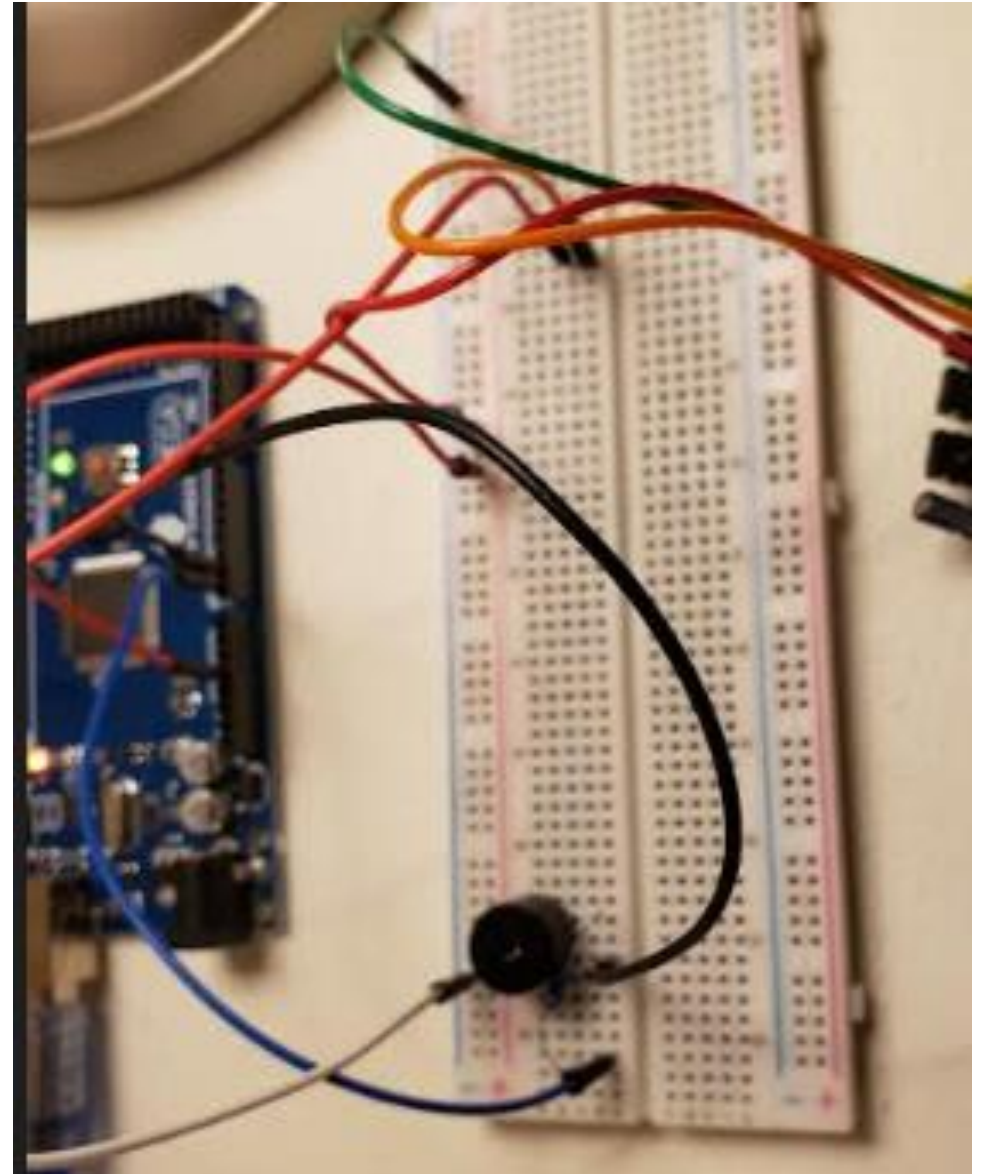
  }
  else {
    digitalWrite(buzzerPin, LOW); // turn Buzzer OFF
  }
}
```

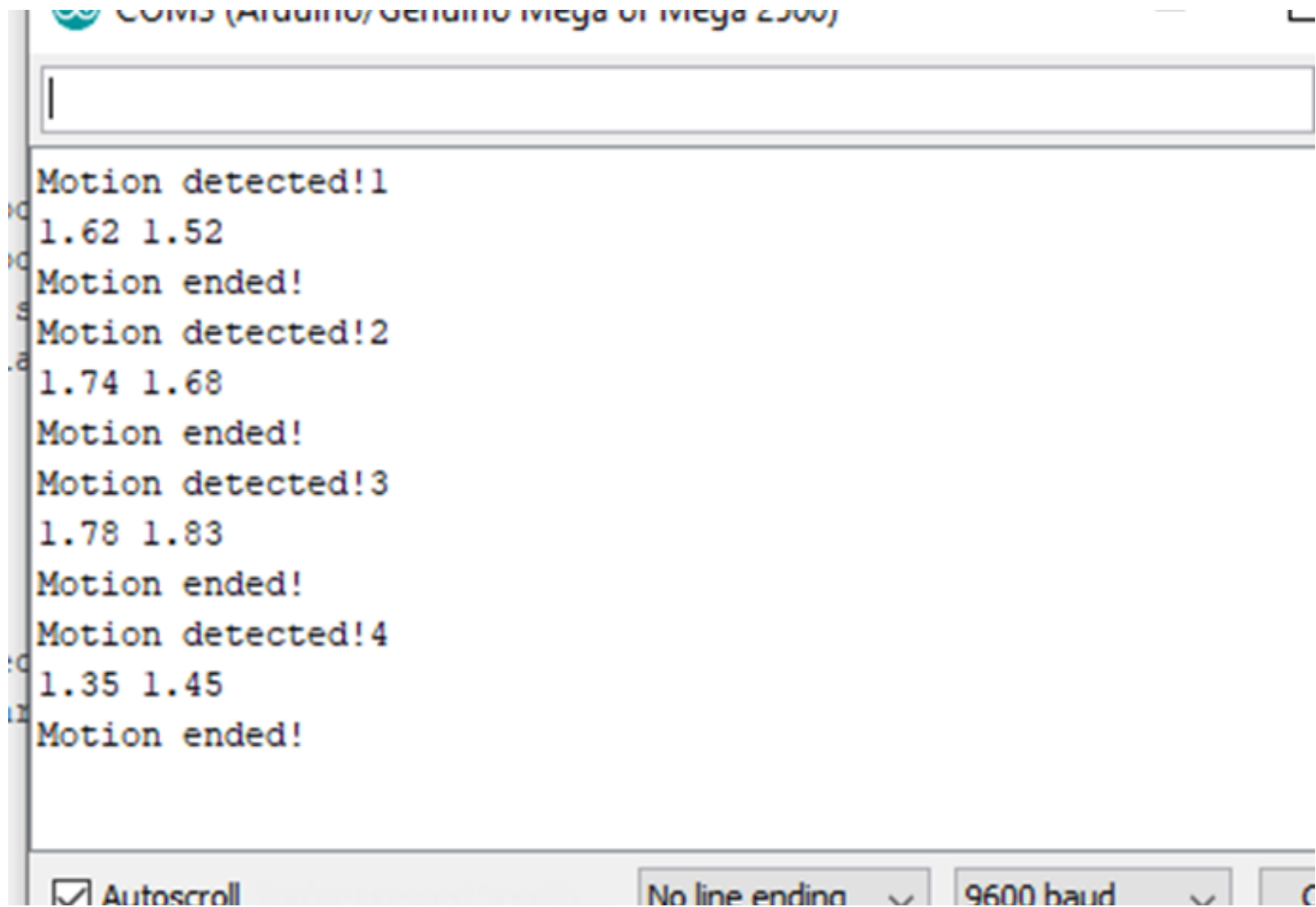


Code Explanation

- The code defines pin 7 as the input and pin 8 as the output.
 - These pins can be changed by editing the code.
 - The Arduino setup() function will run one time setting the input and output pins.
-

Motion Sensor w/resistor circuit





The screenshot shows the Arduino IDE Serial Monitor window. The title bar reads "COM5 (Arduino/Genuino Mega or Mega 2560)". The main text area displays the following output:

```
Motion detected!1  
1.62 1.52  
Motion ended!  
Motion detected!2  
1.74 1.68  
Motion ended!  
Motion detected!3  
1.78 1.83  
Motion ended!  
Motion detected!4  
1.35 1.45  
Motion ended!
```

At the bottom, the settings bar includes a checked "Autoscroll" checkbox, a "No line ending" dropdown menu, a "9600 baud" dropdown menu, and a "C" button.

Motion
Detector
w/Serial
Monitor
w/Voltage
Count Reading



Data Display

- The code that displays motion detection count: `count=count+1;`
 - This says that the next display will be plus one.
-
- Viewing data on the serial monitor can give you information about your circuit.
 - It reports the voltage reading and it advised us when motion was detected and how many times.
-

● devry

Wi-Fi Name (SSID)

devry

Wi-Fi Security

WPA2-PSK

Wi-Fi Key ⓘ

.....



SSID Visibility

Shown

Channel

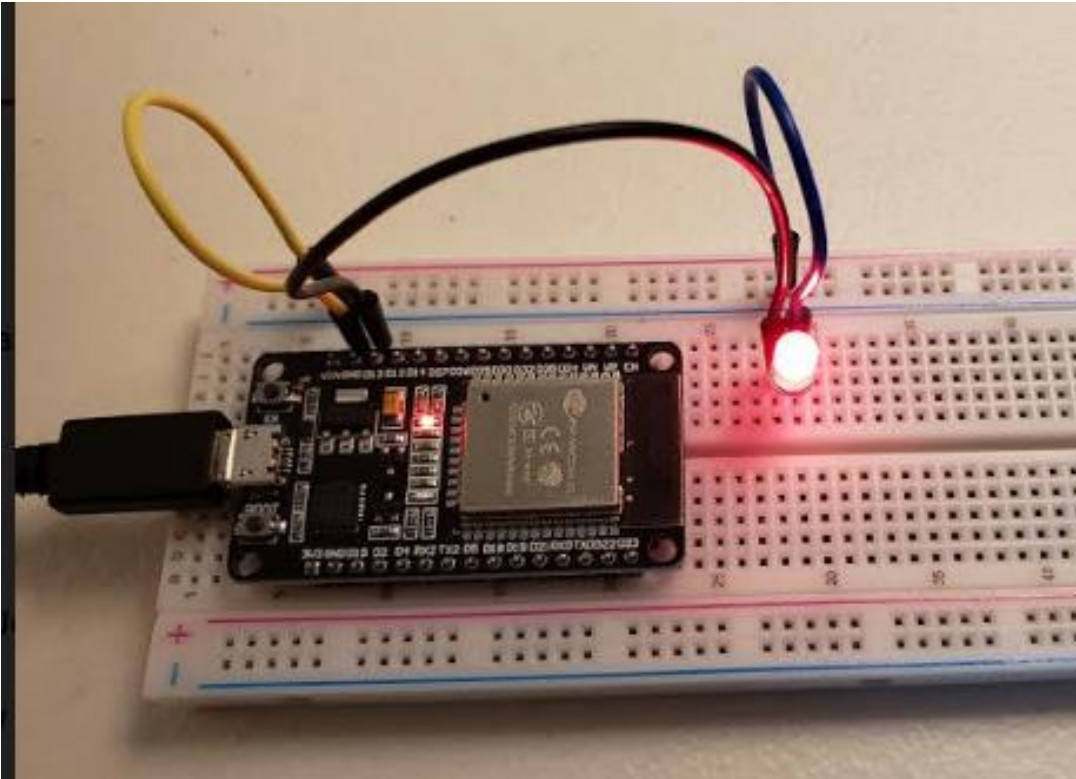
6

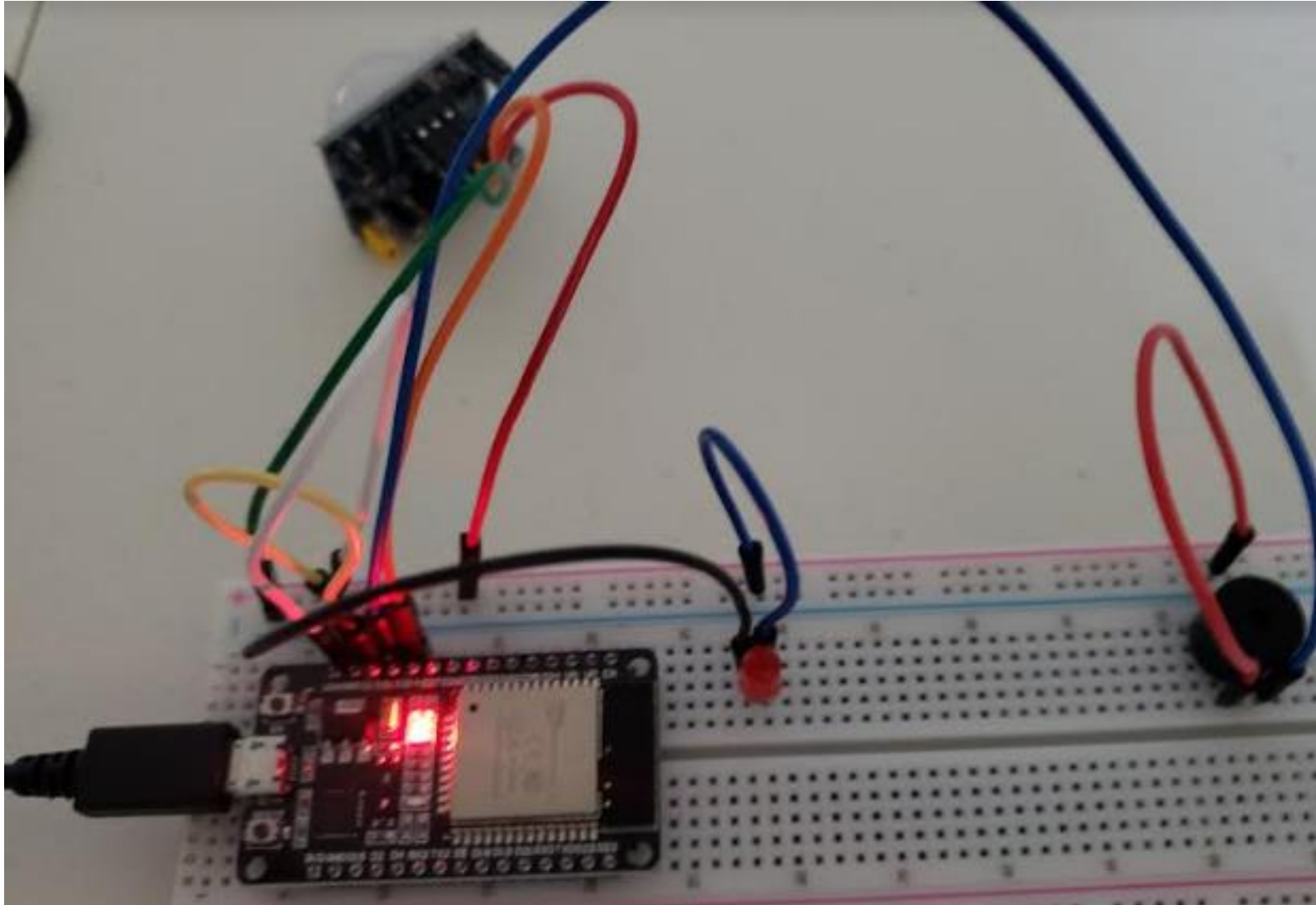
Speed

300M

Router
setup page

ESP32 board w/LED Circuit

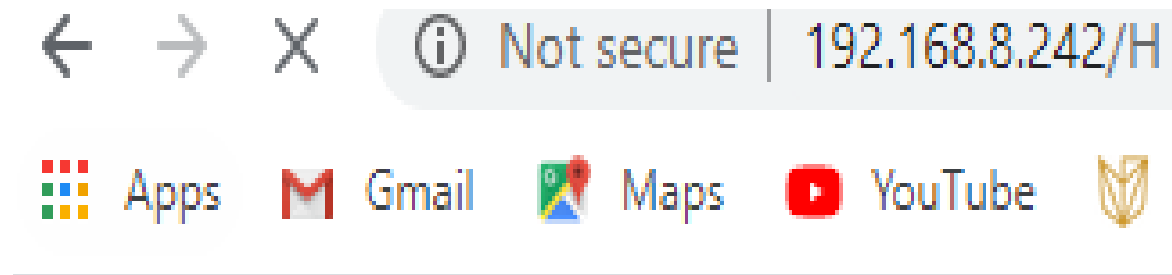




Challenge Assignment:

Motion detector with LED and Buzzer with ESP32

Webpage to turn on/off LED



Welcome to Roger Burns's website!

Click [here](#) to turn the LED on.

Click [here](#) to turn the LED off

Career Skills:

A thick yellow horizontal bar spans the width of the slide, with a vertical yellow bar on the right side.

- Communication – Flowchart was used to forecast the events of the project.
- Electronics – Breadboard, Arduino, ESP32 and work with resistors and voltage was used in this project.
- Programming - The Arduino IDE was utilized.
- Networking – Set up and configured home network.
- Security – Used secure authentication password on network.

Conclusion:

- This course object was to discover the building blocks of the IoT by creating a home security system.
- An Arduino connected to a breadboard and a motion detector was first used.
- Then a ESP32 board was used in place of the Arduino enabling wifi. From here a website was established using code to be able to remotely control the breadboard functions.

