|  |  |  |
| --- | --- | --- |
| **Student Name** |  | **Student Number** |
| Niraj Chaudhary |  | 2332917 |

**Portfolio Introduction**

**Workshop Activities 50% Weighting**

**Mini Project 50% Weighting**

**This completed portfolio will need submitting to Canvas by the due date.**

**Questions please email**

**Dr Sarah Slater**

**s.i.slater@wlv.ac.uk**

**Portfolio**

Contents

[Workbook 1 4](#_Toc105489596)

[Activity 1.1: Actual voltage across 5V breadboard pins. 4](#_Toc105489597)

[Activity 1.2: Actual voltage across 3.3V breadboard pins. 4](#_Toc105489598)

[Activity 1.3: Potential Divider Calculations 4](#_Toc105489599)

[Activity 1.4: 3V Calculations from either the 5V supply or 3.3V supply 5](#_Toc105489600)

[Activity 1.5: Voltage Divider circuit readings from Breadboard circuit. 6](#_Toc105489601)

[Activity 1.6: LED Circuits 6](#_Toc105489602)

[Activity 1.7: Current Measurement 7](#_Toc105489603)

[Activity 1.8: Fritzing for 4 switches & LEDS 9](#_Toc105489604)

[Activity 1.9: Fritzing for Number 0-7 10](#_Toc105489605)

[Workbook 2 11](#_Toc105489606)

[Activity 2.1: LED Flashing to show decimal number 63 as binary. 11](#_Toc105489607)

[Activity 2.2: 4 LED’s for counting up in binary from 0 to 15. 12](#_Toc105489608)

[Activity 2.3: Traffic Lights 15](#_Toc105489609)

[Workbook 3 17](#_Toc105489610)

[Activity 3.1: Circuit Diagram of Button & LED 17](#_Toc105489611)

[Activity 3.2: 3 Switches & Led 19](#_Toc105489612)

[Activity 3.3: 8 Buttons & LEDs (SWITCH STATEMENTS) 21](#_Toc105489613)

[Workbook 4 23](#_Toc105489614)

[Activity 4.1: Serial Port 23](#_Toc105489615)

[Activity 4.2: Serial Port binary to decimal 26](#_Toc105489616)

[Activity 4.3: Calibrating Analogue Information 28](#_Toc105489617)

[Activity 4.4: Temperature Sensor & Serial Port 31](#_Toc105489618)

[Workbook 5 33](#_Toc105489619)

[Activity 5.1: RGB Led and switches 33](#_Toc105489620)

[Activity 5.2: Distance Sensor 35](#_Toc105489621)

[Activity 5.3: 1602 LCD Display 37](#_Toc105489622)

[Workbook 6 40](#_Toc105489623)

[Activity 6.1: PWM 40](#_Toc105489624)

[Workbook 7 42](#_Toc105489625)

[Activity 7.1: 2 Arduinos – using Digital Pins 42](#_Toc105489626)

[Activity 7.2: 2 Arduinos – using Serial I/O 44](#_Toc105489627)

[Workbook 8 46](#_Toc105489628)

[Activity 8.1: Stepper Motor Circuit Diagram 46](#_Toc105489629)

[Activity 8.2: 2 Stepper Motors 48](#_Toc105489630)

[Workbook 9 49](#_Toc105489631)

[Activity 9.1: Windscreen Wiper Code using Servos & Temperature Sensor 49](#_Toc105489632)

[Individual Project (50%) 51](#_Toc105489633)

[Rationale 51](#_Toc105489634)

[Timescales 51](#_Toc105489635)

[Equipment 51](#_Toc105489636)

[The Project 51](#_Toc105489637)

[Step 1 produce adetailed description of your project. 51](#_Toc105489638)

[Step 2 Circuit Diagram&Fritzing Schematic 51](#_Toc105489639)

[Step 3 A Program 52](#_Toc105489640)

[Step 4 Testing 52](#_Toc105489641)

[Step 5 Conclusions 52](#_Toc105489642)

[Layout 52](#_Toc105489643)

[Marking 52](#_Toc105489644)

[All sections carry equal marks. 52](#_Toc105489645)

If you prefer, you may use Tinkercad to show a component layout, rather than a circuit Diagram in Fritzing or other circuit design software, though a circuit diagram is more useful as this is what you would most likely see if you were working on embedded systems.

# Workbook 1

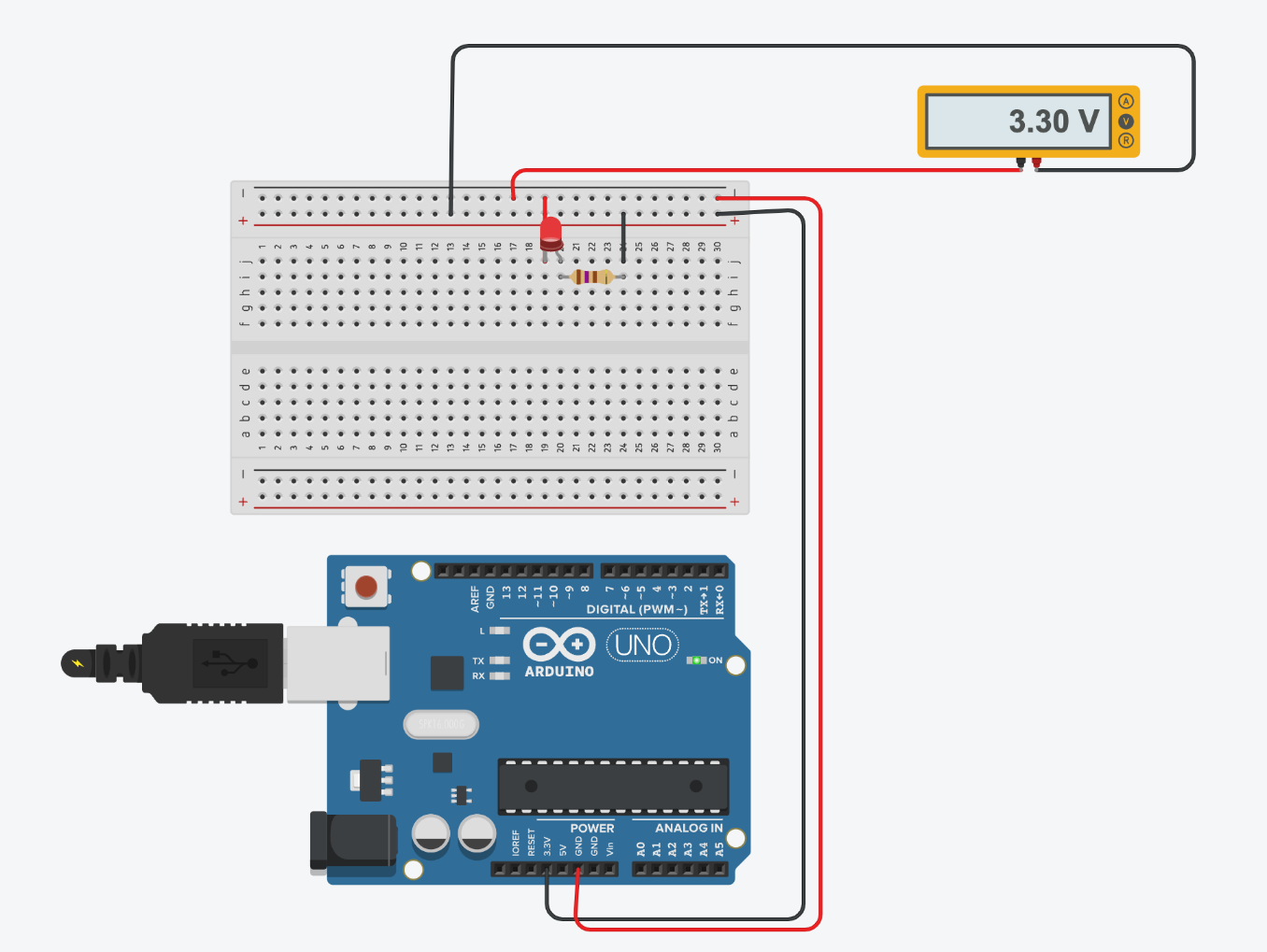
## Activity 1.1: Actual voltage across 5V breadboard pins.

5.00 V

# 

## Activity 1.2: Actual voltage across 3.3V breadboard pins.

3.30 V



Explain in around 100 words why you think the value read by a multi meter on a circuit, may be different to a simulator value such as TinkerCad.

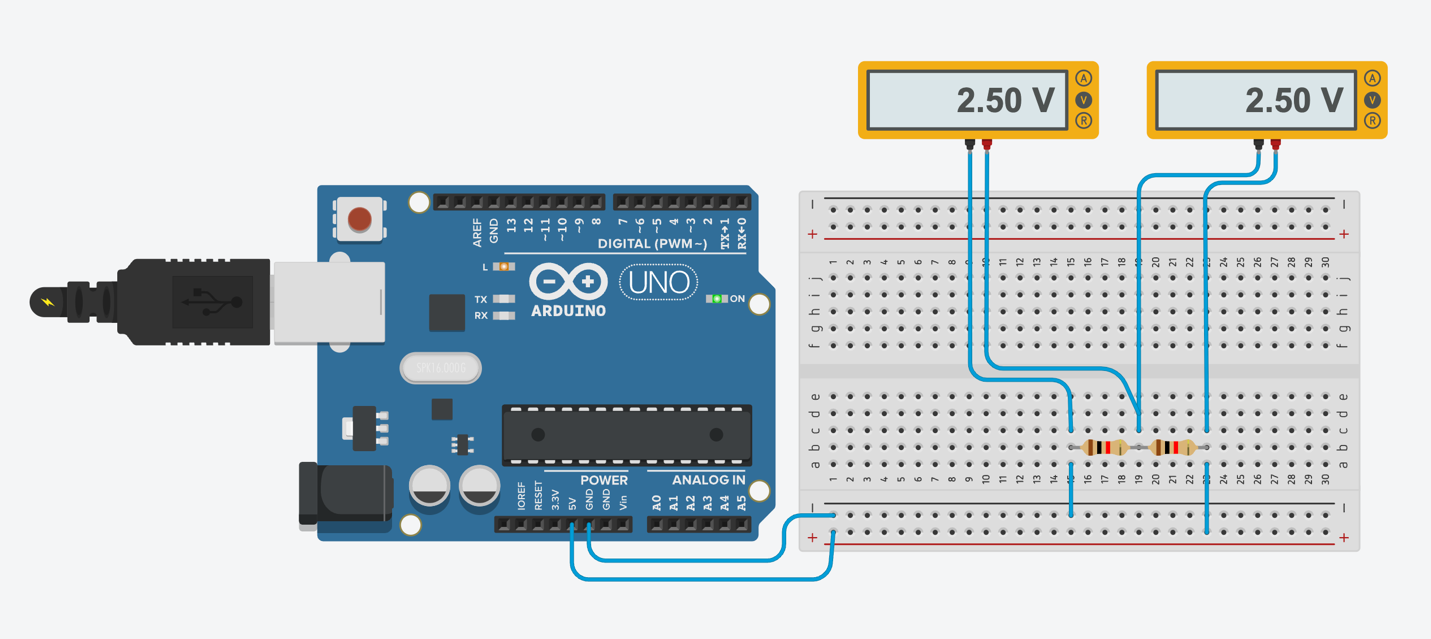
* Due to environmental conditions, the value gotten from a multimeter on a circuit may not match the value obtained from a simulant like TinkerCad. The value may vary depending on where we are consuming the circuit and the surrounding environment, including the ambient temperature where the hardware is functioning. The age of the hardware items may also contribute to a value that differs from a simulator. The value may fluctuate from a simulator depending on how frequently and how long the hardware is utilized. Resistance, capacitance, and conductivity are a few examples of features that differ depending on the component's temperature. The individualities of these components can change when the surrounding temperature changes, changing the recorded results. For time, the behavior of the circuit as a whole may change dependent on whether a resistor's resistance increases or decreases with temperature.

If the read value is 4.84V on a 5V supply, what would be a sensible tolerance to quote, explain your answer.

The tolerance for electronic equipment can be estimated at +-5%, and the range of the voltage fluctuation is between 4.75 and 5.25 volts, thus the voltage can change.

## Activity 1.3: Potential Divider Calculations

Show the working on how you achieved 2.5V



## Activity 1.4: 3V Calculations from either the 5V supply or 3.3V supply

Solution

Voltage (V) = 5V

Voltage Drop (VD) = 3V

Voltage In (Vin) = 5v

Voltage Out (Vout) = 3v

1st Resistor (R1) = 220 ohm

2nd Resistor (R2) = ?

Total Resistance (R) = R1 + R2 = 220 + x (Let R2 be ‘x’)

Voltage Drop (VD) = (R2/R) X 5

Or, 3 = ((x/220+x)) X 5

Or, 3 = 5x/220+x

Or, 3(220 + x) = 5x

Or, 660 + 3x = 5x

Or, 660 = 5x – 3x

Or, 660 = 2x

x = 330 ohm

330 ohm - 220 ohm = 110 ohm

1/R = 1/220 + 1/220

Or, 1/R = 2/220

Or, 1/R = 1/110

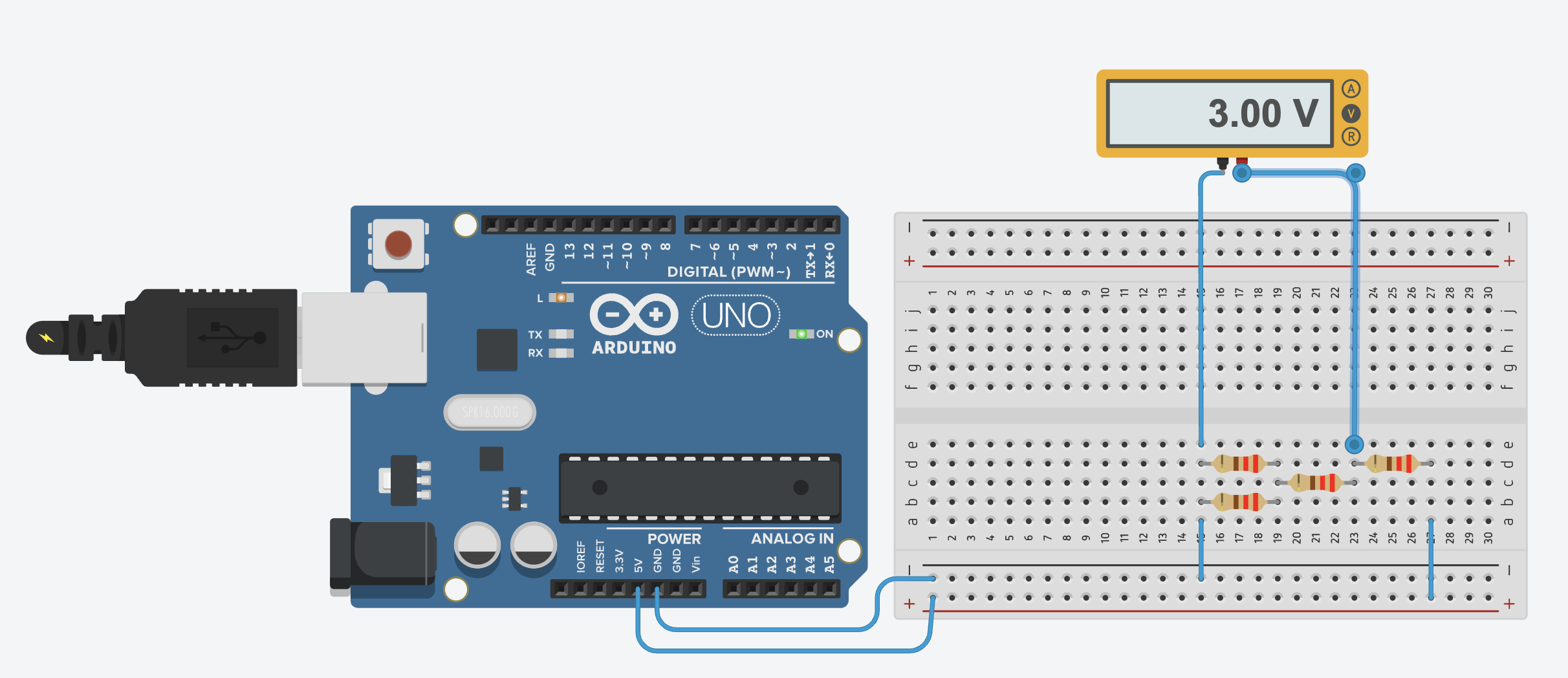
R = 110 ohm

Vout = Vin \* (R2/(R1+R2))

Or, 3 = 5 \* (10/10 + R2)

R2 = 6.67k ohm

## Activity 1.5: Voltage Divider circuit readings from Breadboard circuit.



## Activity 1.6: LED Circuits

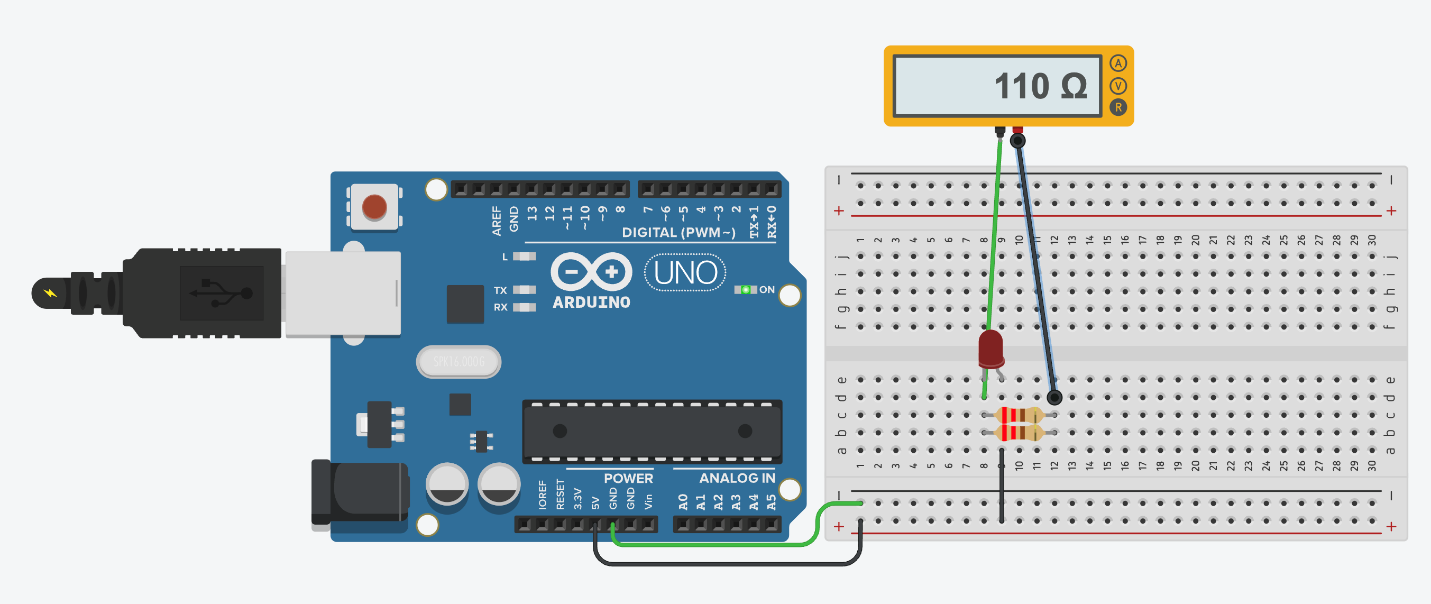
Diagram

Description automatically generated

Each resistor Value

220 0hm

220 ohm



Total resistance Calculation

Solution

1/R =1/R1 +1/R2

1/R = 1/220 + 1/220

1/R = 2/220

R = 110ohm

Measured Resistance

110 ohm

If measured resistance is not the same, why not? If you simulated this, why might the real value be different.

The hardware made has environmental elements like room temperature which can cause the circuit to behave and the values/resistance can change from the values/resistance on a simulator, therefore the circuit made on a simulator and made using hardware is different and offers various outputs and values.

## Activity 1.7: Current Measurement

Calculation of current flowing into LED

Solution

Current of the LED = 26.1/1000 = 0.0261

Voltage (V) = 5 V

Voltage (V) = Voltage Source – Voltage Forward

= 5 - 2.5 = 2.5 V

Resistance (R) = ?

Current (I) = ?

1/R = 1/R1 + 1/R2

Or, 1/R = 1/220 + 1/220

Or, 1/R = 1/110

R = 110 ohm

V = IR

Or, 2.5 = I X 110

Or, I = 2.5/110

I = 0.02272727273

Actual measured value of current

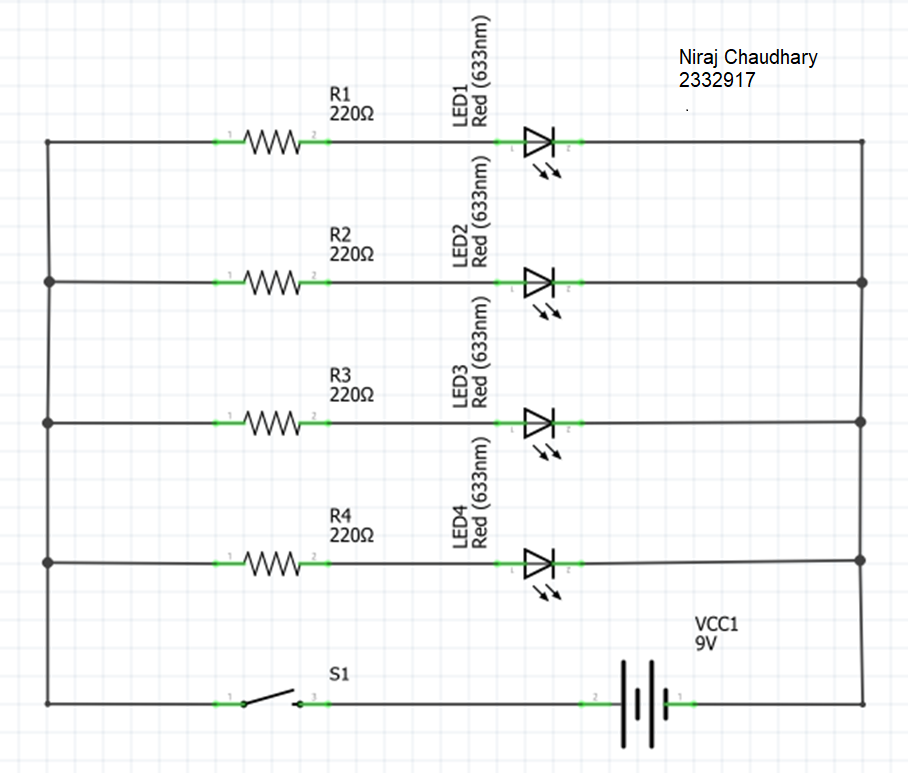
0.045 A

Why might they be different?

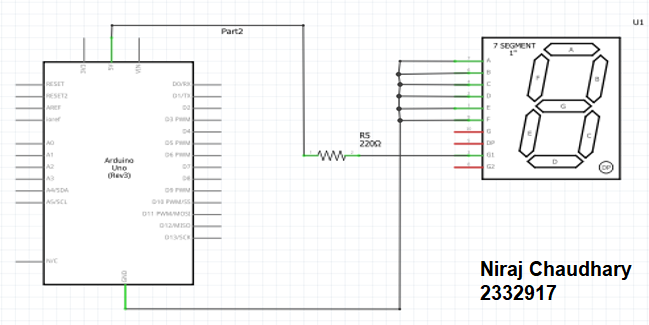
Ans : Because the current in the circuit is 45 mA, whereas the current coming from the led is 26.1 mA.

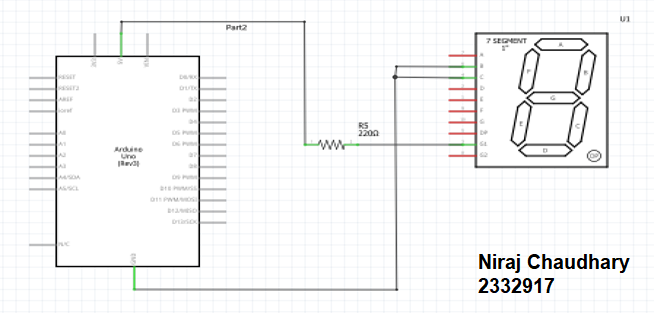


## Activity 1.8: Fritzing for 4 switches & LEDS

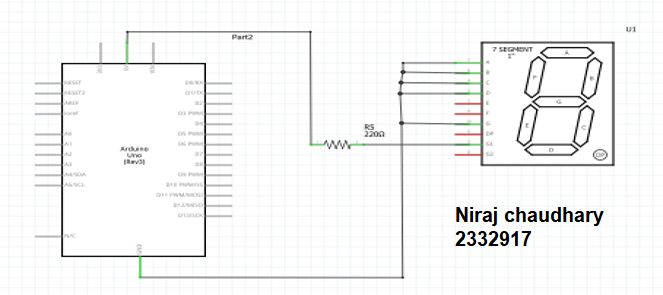


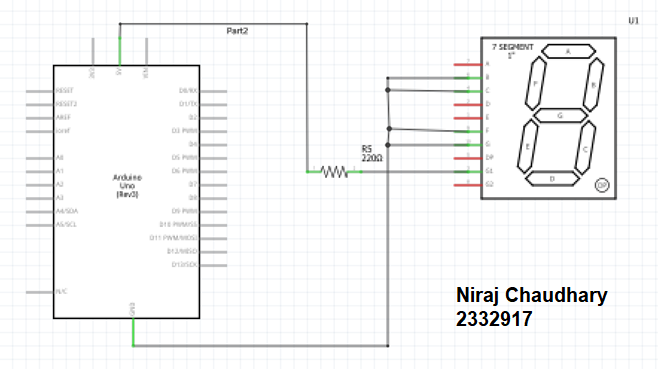
# Activity 1.9: Fritzing for Number 0-7

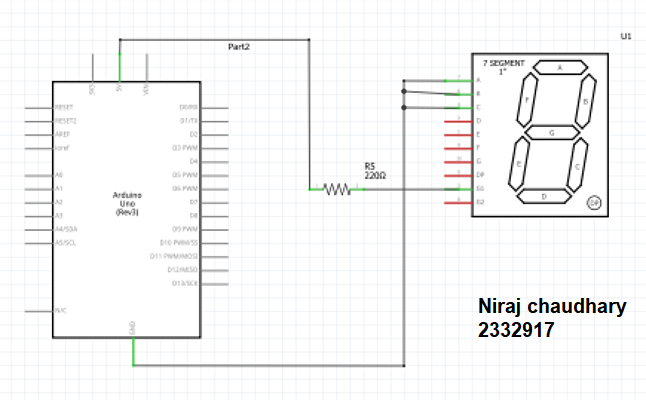
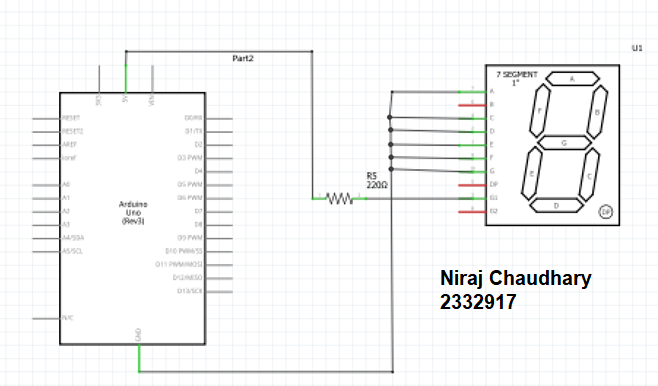
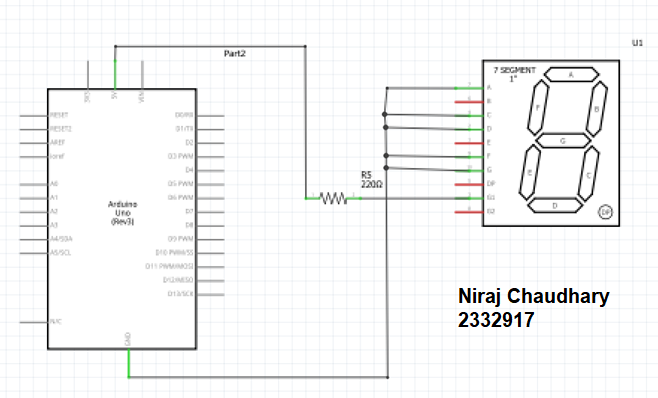








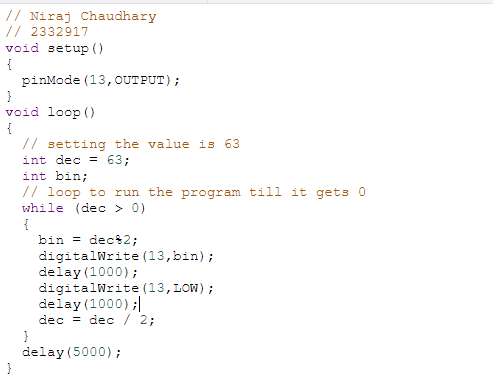




# Workbook 2

## Activity 2.1: LED Flashing to show decimal number 63 as binary.

63 as binary, including working



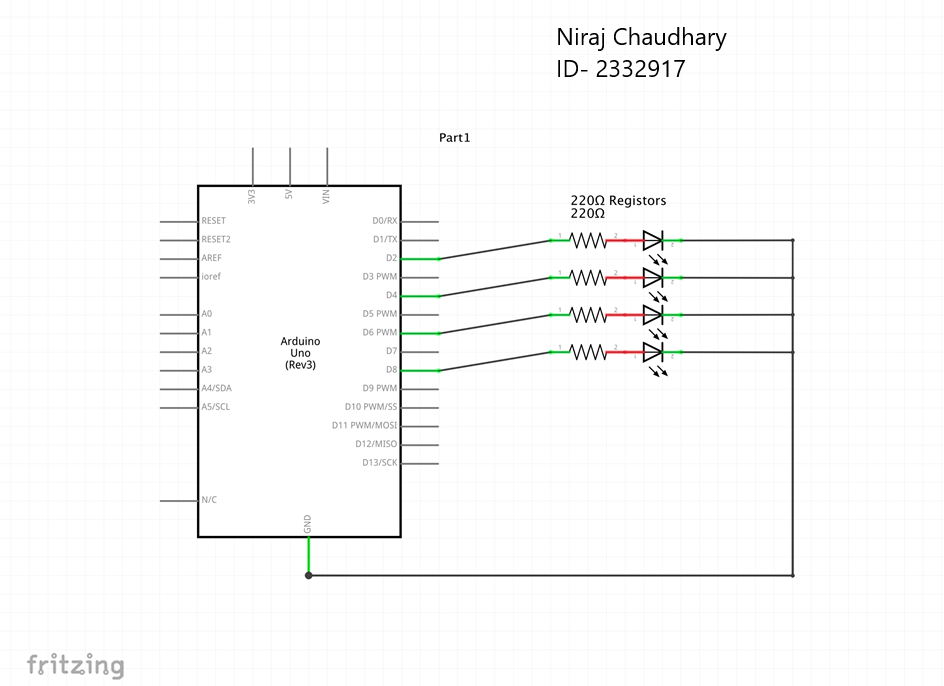
Copy & Post your code with a suitable comment at the top of code with your name & student number ☺

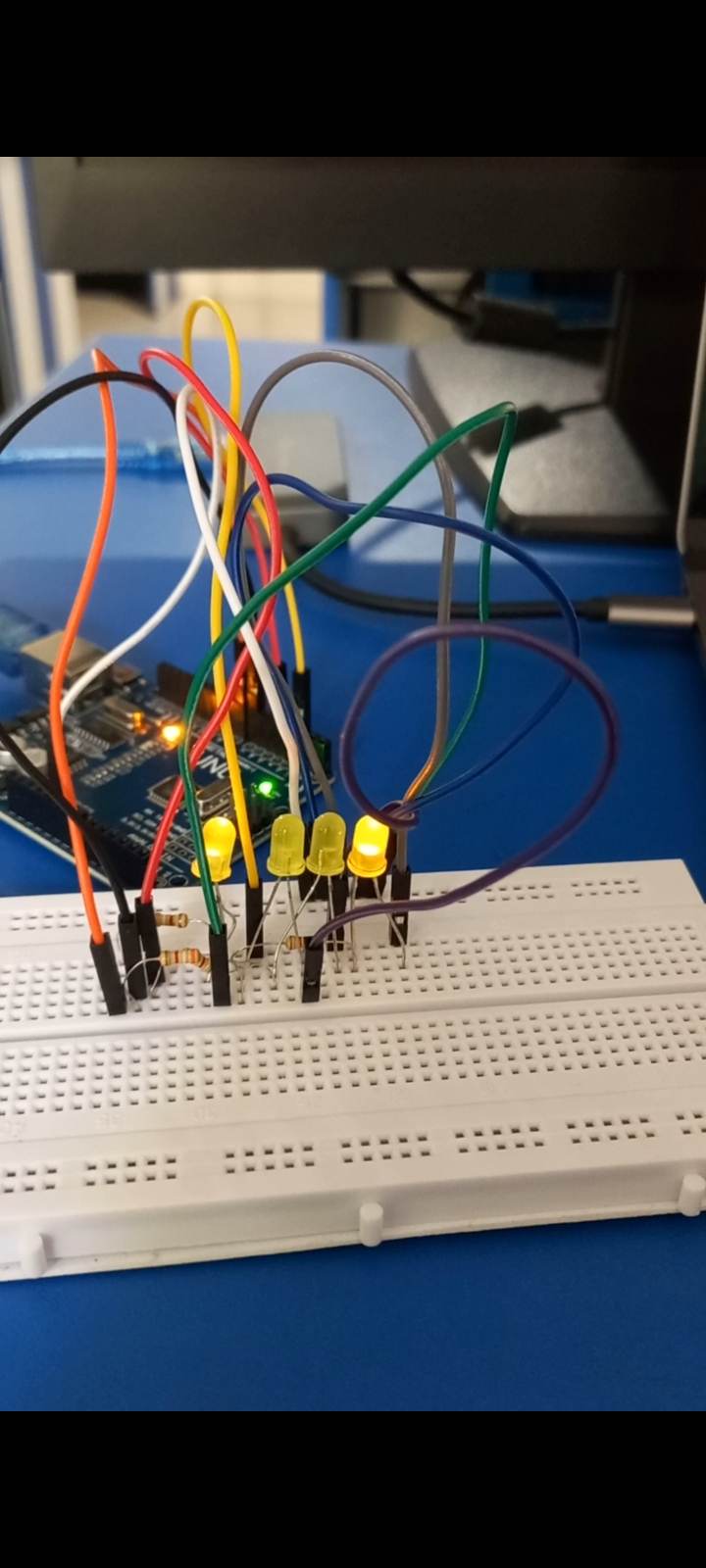
/\*Niraj Chaudhary

/\*2332917  
 program to show decimal number 63 as binary \*/  
void setup()   
{   
 pinMode(8, OUTPUT);   
}  
void loop()  
{   
 int num = 63;  
 int bin;   
//converting 63 into binary  
 while (num!=0)  
{  
 bin = num%2;  
 digitalWrite(8, bin);   
delay(1000);   
// Wait for 1000 millisecond  
 bin = 0;   
 digitalWrite(8,bin);  
 delay(1000);  
 num = num/2;  
}  
 delay(1000);   
}

## Activity 2.2: 4 LED’s for counting up in binary from 0 to 15.

Fritzing Circuit diagram for Step 4 i.e. 4 LEDs





Arduino Program for Step 4 i.e. 4 LEDs

// Niraj Chaudhary

// 2332917

int pin2 = 2;

int pin4 = 4;

int pin7 = 7;

int pin8 = 8;

void setup(){

pinMode(pin2,OUTPUT);

pinMode(pin4,OUTPUT);

pinMode(pin7,OUTPUT);

pinMode(pin8,OUTPUT);

}

void loop(){

{

//FOR 0000 - 0

digitalWrite(pin2,LOW);

digitalWrite(pin4,LOW);

digitalWrite(pin7,LOW);

digitalWrite(pin8,LOW);

delay(2000);

//FOR 0001 - 1

digitalWrite(pin2, HIGH);

delay(1000);

//FOR 0010 - 2

digitalWrite(pin2,LOW);

digitalWrite(pin4, HIGH);

delay(1000); // Wait for 1000 millisecond(s)

//FOR 0011 - 3

digitalWrite(pin2,HIGH);

digitalWrite(pin4, HIGH);

delay(1000); // Wait for 1000 millisecond(s)

//FOR 0100 - 4

digitalWrite(pin7,HIGH);

digitalWrite(pin4,LOW);

digitalWrite(pin2,LOW);

delay(1000);

//FOR 0101 - 5

digitalWrite(pin4,LOW);

digitalWrite(pin2,HIGH);

delay(1000);

//FOR 0110 - 6

digitalWrite(pin7,HIGH);

digitalWrite(pin4,HIGH);

digitalWrite(pin2,LOW);

delay(1000);

//FOR 0111 - 7

digitalWrite(pin2,1);

delay(1000);

//FOR 1000 - 8

digitalWrite(pin8,1);

digitalWrite(pin2,0);

digitalWrite(pin7,0);

digitalWrite(pin4,0);

delay(1000);

//FOR 1001 - 9

digitalWrite(pin8,1);

digitalWrite(pin2,1);

digitalWrite(pin7,0);

digitalWrite(pin4,0);

delay(1000);

//FOR 1001 - 10

digitalWrite(pin8,1);

digitalWrite(pin2,0);

digitalWrite(pin7,0);

digitalWrite(pin4,1);

delay(1000);

//FOR 1011 - 11

digitalWrite(pin8,1);

digitalWrite(pin2,1);

digitalWrite(pin7,0);

digitalWrite(pin4,1);

delay(1000);

//FOR 1100 - 12

digitalWrite(pin8,1);

digitalWrite(pin2,0);

digitalWrite(pin7,1);

digitalWrite(pin4,0);

delay(1000);

//FOR 1101 - 13

digitalWrite(pin8,1);

digitalWrite(pin2,1);

digitalWrite(pin7,1);

digitalWrite(pin4,0);

delay(1000);

//FOR 1110 - 14

digitalWrite(pin8,1);

digitalWrite(pin2,0);

digitalWrite(pin7,1);

digitalWrite(pin4,1);

delay(1000);

//FOR 1111 - 15

digitalWrite(pin8,1);

digitalWrite(pin2,1);

digitalWrite(pin7,1);

digitalWrite(pin4,1);

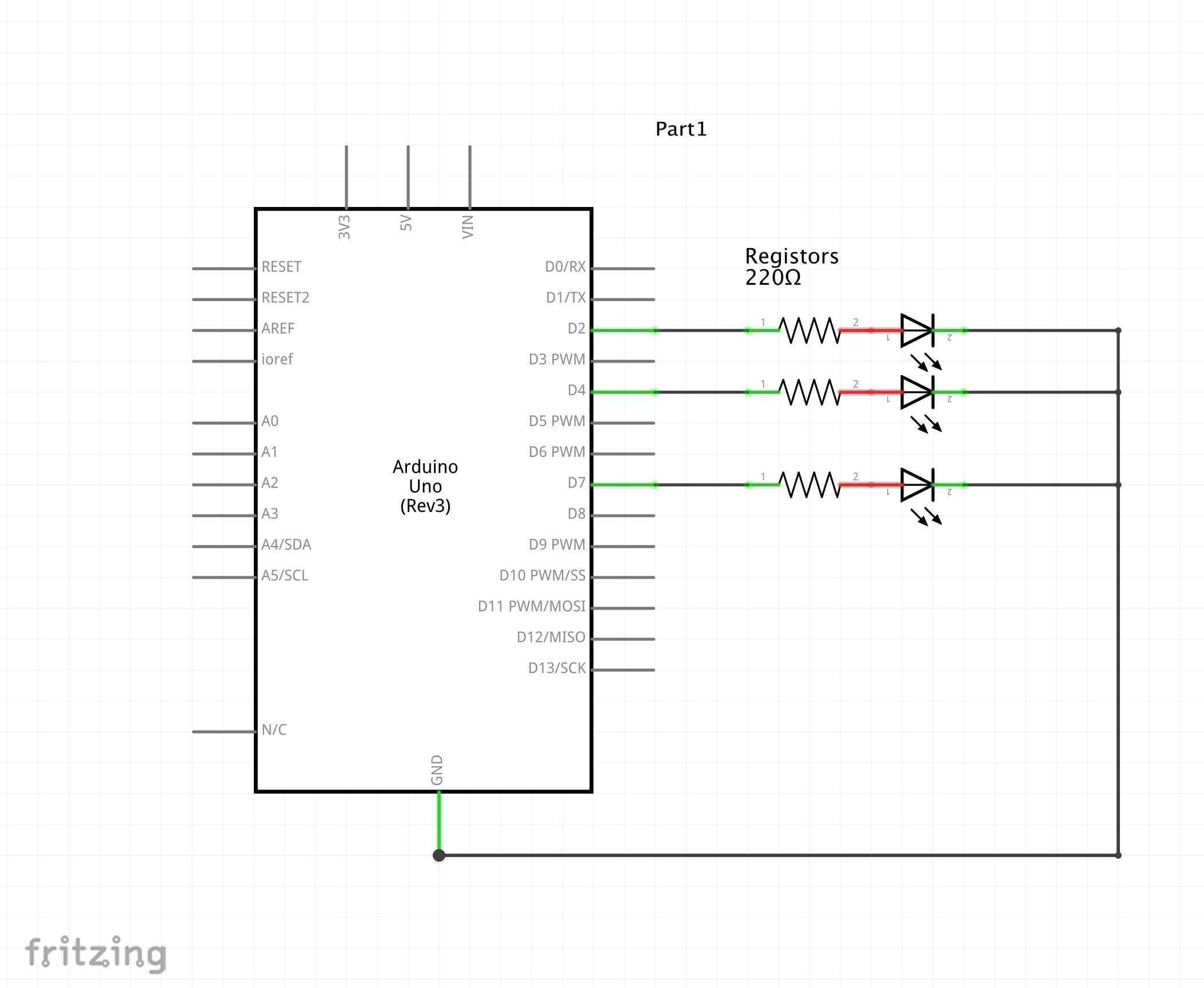
delay(5000);

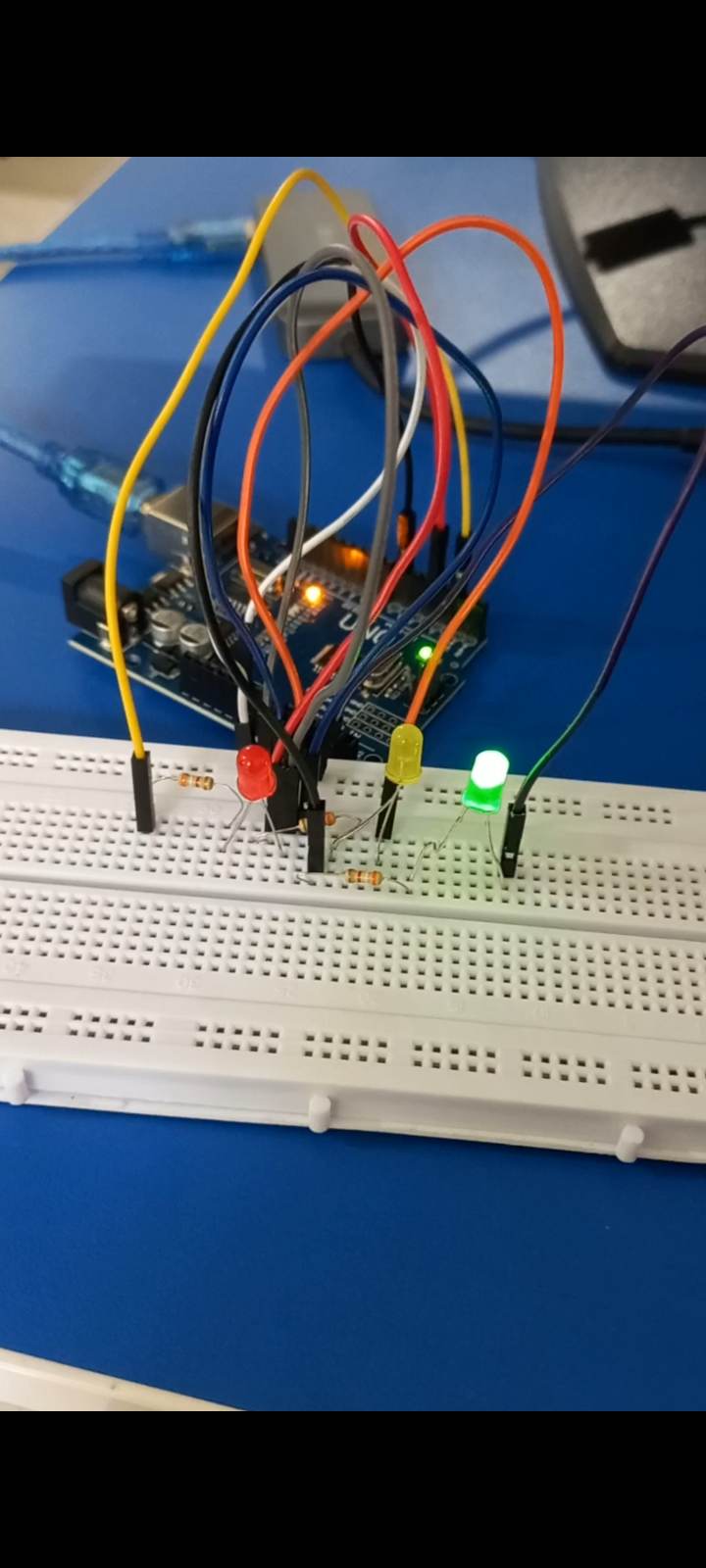
}

}

## Activity 2.3: Traffic Lights

Fritzing Circuit diagram for Step 4 i.e. 4 LEDs





Arduino Program for Step 4 i.e. 4 LEDs

//Made by Niraj Chaudhary

//2332917

int pin2 = 2;

int pin4 = 4;

int pin7 = 7;

void setup(){

//Setting the pin to output mode

pinMode(pin2,OUTPUT);

pinMode(pin4,OUTPUT);

pinMode(pin7,OUTPUT);

}

void loop(){

{

int i = 0; //Initializing i as 0

do { //Using While Loop

digitalWrite(pin2,1); //Pin 2 becomes high

delay(2000); //Delay for 2 seconds.

digitalWrite(pin2,1); //Pin 2 becomes high

digitalWrite(pin4,1); //Pin 4 becomes high

delay(2000); //Delay for 2 seconds

digitalWrite(pin2,0); //Pin 2 becomes low

digitalWrite(pin4,0); //Pin 4 becomes low

digitalWrite(pin7,1); //Pin 7 becomes high

delay(2000); //Delay for 2 seconds

digitalWrite(pin2,0); //Pin 2 becomes low

digitalWrite(pin4,1); //Pin 4 becomes high

digitalWrite(pin7,0); //Pin 7 becomes low

delay(2000); //Delay for 2 seconds.

digitalWrite(pin2,0); //Pin 2 becomes low

digitalWrite(pin4,0); //Pin 4 becomes low

digitalWrite(pin7,0); //Pin 7 becomes low

delay(2000); //Delay for 2 seconds

i++; // Increase the value of i by 1

} while (i < 5); //Whenever i < 5

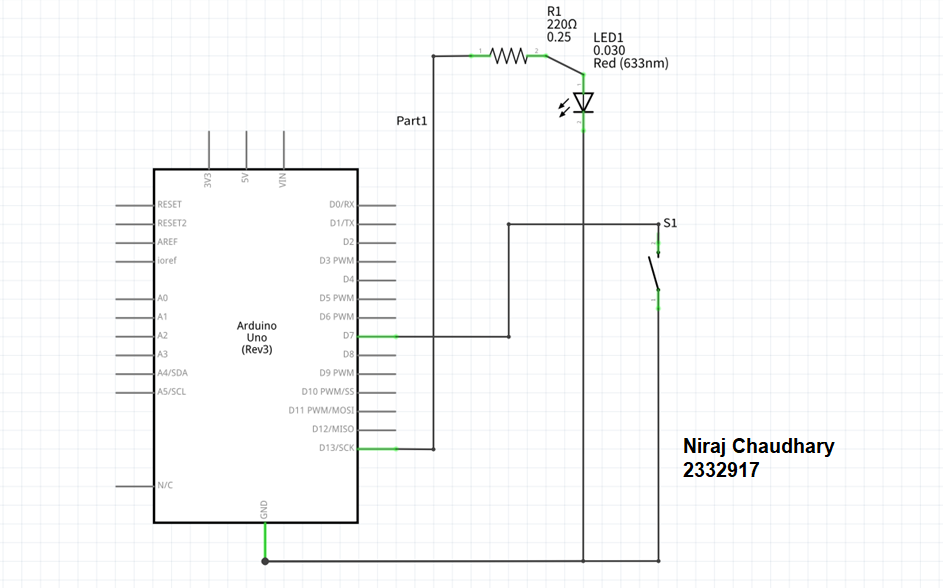
delay(5000); //Delaying for 5 seconds after end of 5 iterations.

}

# Workbook 3

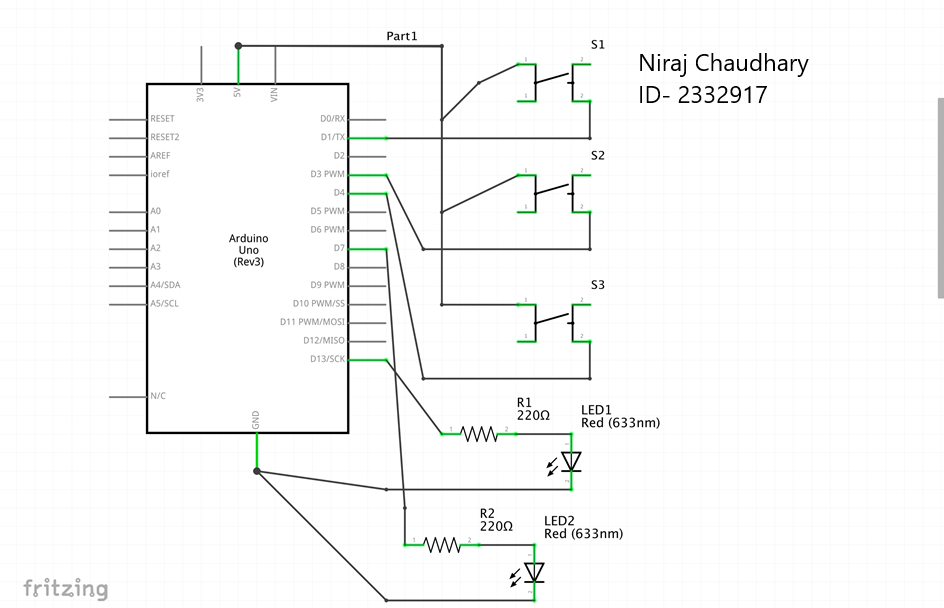
## Activity 3.1: Circuit Diagram of Button & LED

**Fritzing**



## Activity 3.2: 3 Switches & Led

Fritzing Circuit Diagram



Arduino Program

//Niraj Chaudhary

//2332917

//Initializing the pins and led

int rLed = 13;

int yLed = 7;

int buttonPin = 1;

int buttonPin2 = 3;

int buttonPin3 = 4;

//Setting up the input an the output mode.

void setup(){

pinMode(rLed,OUTPUT);

pinMode(buttonPin,INPUT);

pinMode(yLed,OUTPUT);

pinMode(buttonPin2,INPUT);

pinMode(buttonPin3,INPUT);

}

void loop(){

//When the first button is clicked

if (digitalRead(buttonPin)==HIGH)

{

//Red led lights for 1 seconds

digitalWrite(rLed, HIGH);

delay(1000);

}

else

{

//Else, it is low

digitalWrite(rLed, LOW);

}

}

//When the second button is clicked.

if (digitalRead(buttonPin2)==HIGH)

{

//Yellow led lights for 2 seconds

digitalWrite(yLed, HIGH);

delay(2000);

}

else

{

//Else, it is low

digitalWrite(yLed, LOW);

}

//WHen the third button is clicked

if (digitalRead(buttonPin3)==HIGH)

{

//Red led lights for 3 seconds

digitalWrite(rLed, HIGH);

delay(3000);

}

else

{

//Else, it is low

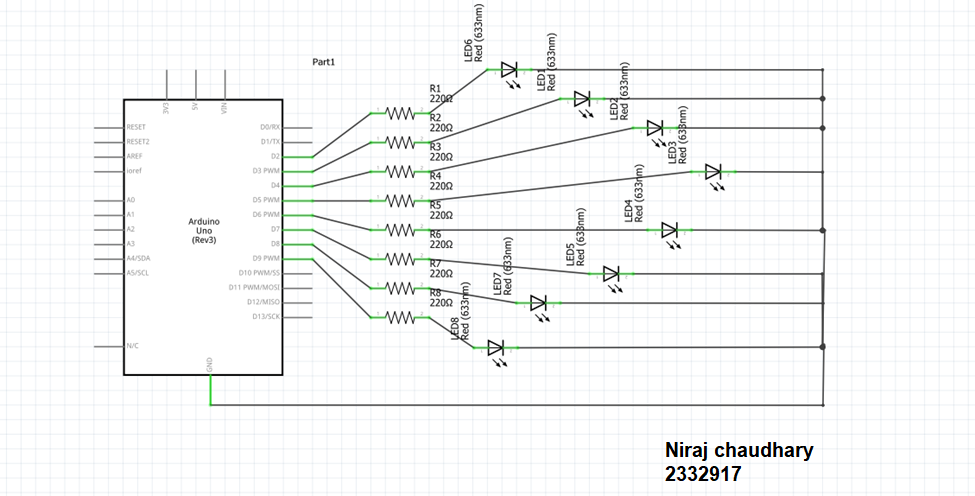
digitalWrite(rLed, LOW);

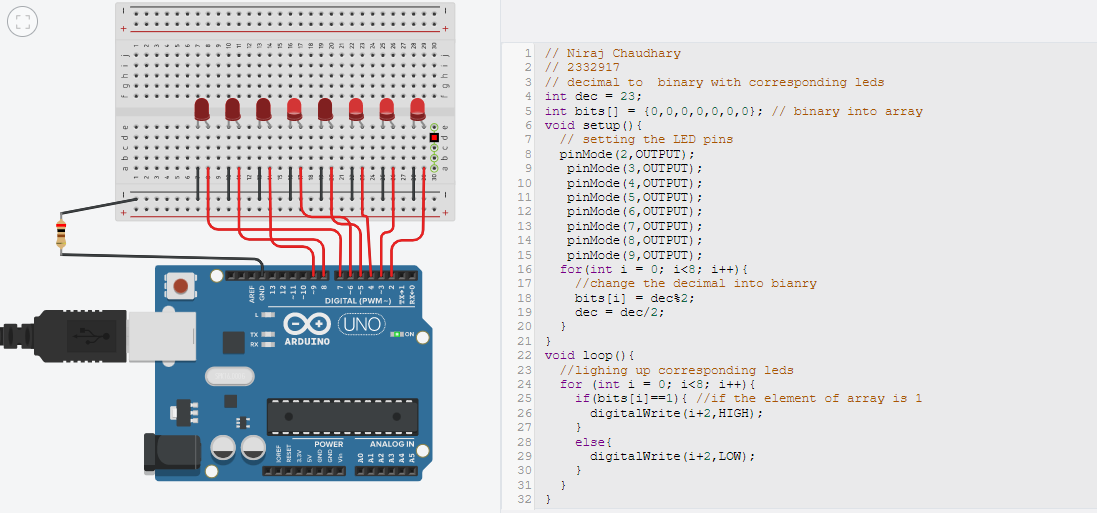
}

}

## Activity 3.3: 8 Buttons & LEDs (SWITCH STATEMENTS)

Fritzing





**Arduino Program**

// Niraj Chaudhary

// 2332917

// decimal to binary with corresponding leds

int dec = 23;

int bits[] = {0,0,0,0,0,0,0}; // binary into array

void setup(){

// setting the LED pins

pinMode(2,OUTPUT);

pinMode(3,OUTPUT);

pinMode(4,OUTPUT);

pinMode(5,OUTPUT);

pinMode(6,OUTPUT);

pinMode(7,OUTPUT);

pinMode(8,OUTPUT);

pinMode(9,OUTPUT);

for(int i = 0; i<8; i++){

//change the decimal into bianry

bits[i] = dec%2;

dec = dec/2;

}

}

void loop(){

//lighing up corresponding leds

for (int i = 0; i<8; i++){

if(bits[i]==1){ //if the element of array is 1

digitalWrite(i+2,HIGH);

}

else{

digitalWrite(i+2,LOW);

}

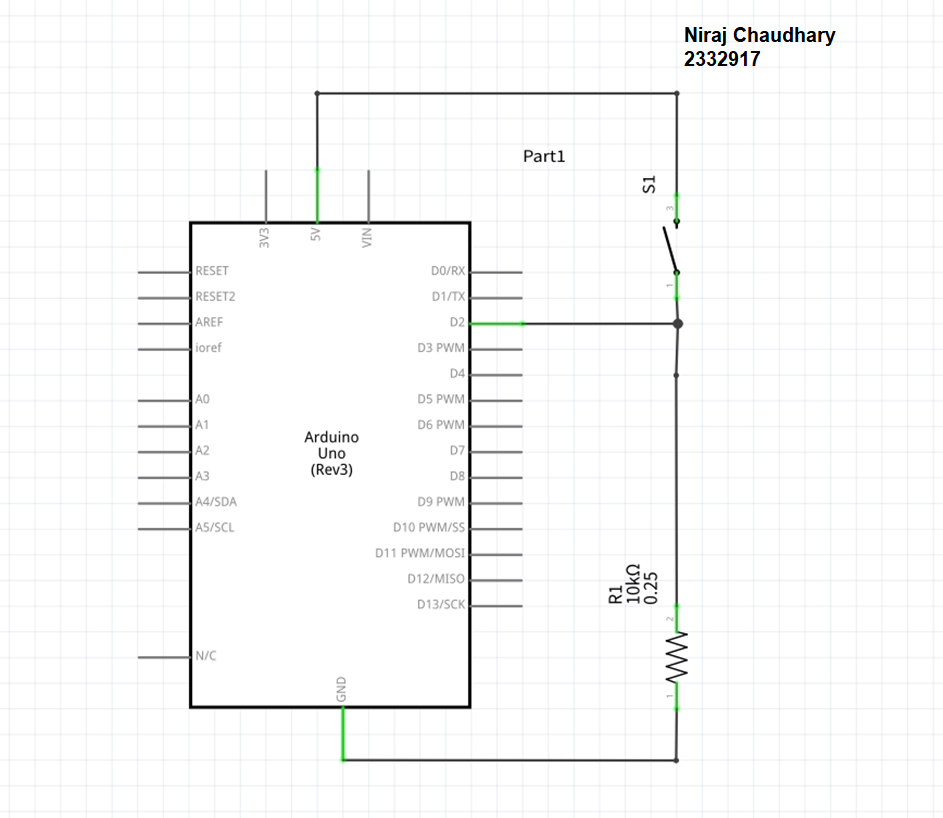
}

}

# Workbook 4

## Activity 4.1: Serial Port

Fritzing



**Arduino Program**

//Niraj Chaudhary

//2332917

//Print name and number into serial monitor

String name = "Niraj Chaudhary";

//Storing id number in string because it is long for an integer

String id = "2332917";

void setup()

{

Serial.begin(9600);

pinMode(13, INPUT);

}

void loop()

{

int switchVal = digitalRead(13);

if( switchVal == 1){

Serial.print(name);

Serial.print('\n');

Serial.print(id);

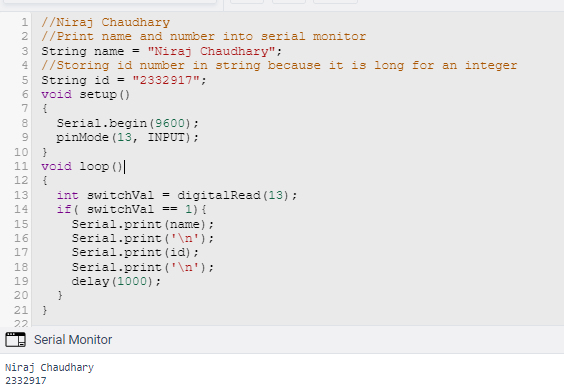
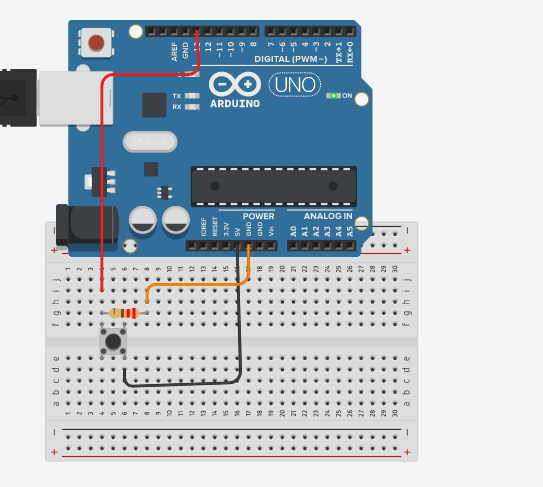
Serial.print('\n');

delay(1000);

}

}

Screen Shot of Serial Port



## Activity 4.2: Serial Port binary to decimal

**Code**

//Niraj Chaudhary

// 2332917

int a = 6,b = 7,c = 8,d = 9,e = 10, f = 11, g = 12, h = 13;

void setup()

{

//setting up the switches

pinMode(a, INPUT);

pinMode(b, INPUT);

pinMode(c, INPUT);

pinMode(d, INPUT);

pinMode(e, INPUT);

pinMode(f, INPUT);

pinMode(g, INPUT);

pinMode(h, INPUT);

Serial.begin(9600);

}

void loop()

{

//making an array for all of the switches

int arr[8] = {digitalRead(a),digitalRead(b),digitalRead(c),

digitalRead(d),digitalRead(e),digitalRead(f),

digitalRead(g),digitalRead(h)};

float decimal = 0;

//to know the actual decimal number

for (int i = 0; i < 8; i++) {

decimal += arr[i]\*pow(2,i);

//each bit multiplied by power of 2 with index of each bits position

}

Serial.print("Decimal of ");

Serial.print(arr[0]);

Serial.print(arr[1]);

Serial.print(arr[2]);

Serial.print(arr[3]);

Serial.print(arr[4]);

Serial.print(arr[5]);

Serial.print(arr[6]);

Serial.print(arr[7]);

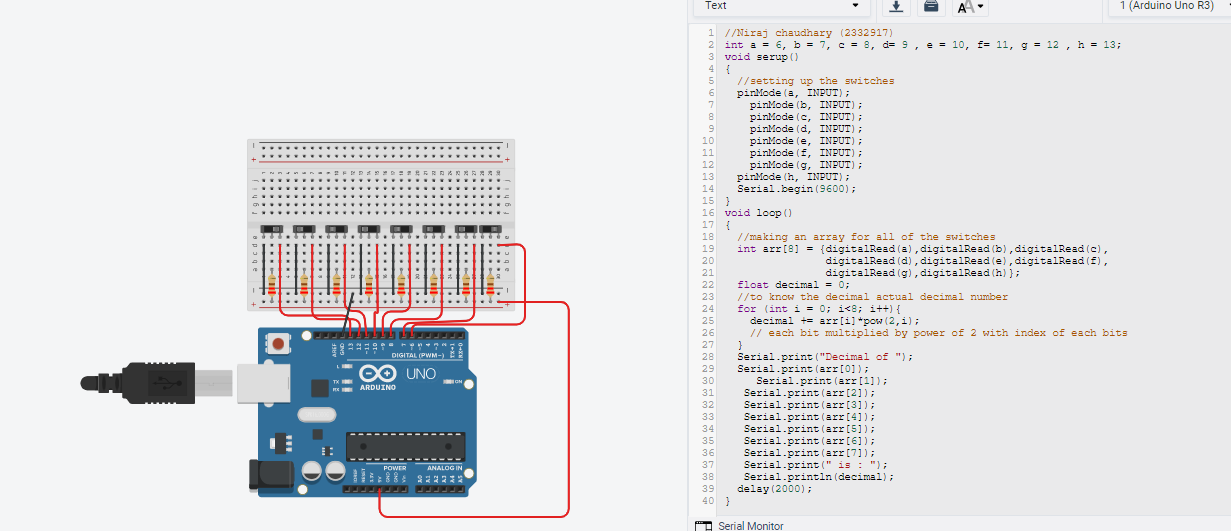
Serial.print(" is: ");

Serial.println(decimal);

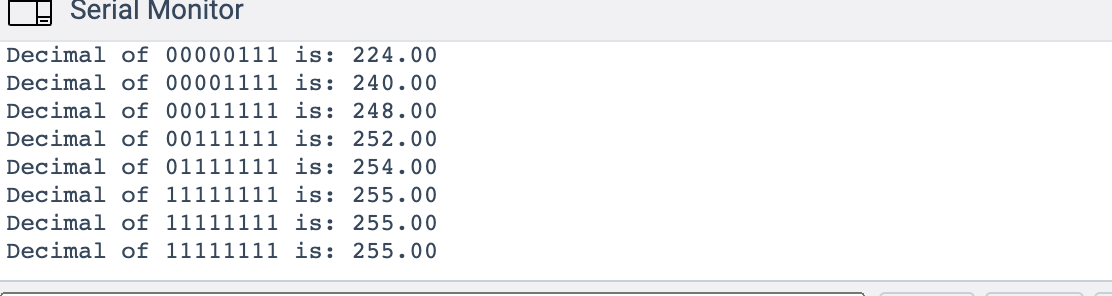
delay(2000);

}

Screen Shot of Serial Port



Serial Monitor Output



## Activity 4.3: Calibrating Analogue Information

Code

//Made By Niraj Chaudhary (2332917)

int buttonPin = 11;

void setup(){

pinMode(buttonPin, INPUT);

Serial.begin(9600);

}

void loop(){

int buttonValue = digitalRead(buttonPin);

if(buttonValue==HIGH){

int value = analogRead(A0);

float voltage = value\*(5.0/1023.0);

Serial.print("the corresponding voltage value is: ");

Serial.println(voltage);

delay(500);

float resistance = voltage\*7.5/5.0;

Serial.print("the resistance in KΩ is: ");

Serial.println(resistance);

delay(500);

}

}

Pot Resistance Clockwise

0.14 KΩ

Pot Resistance Anti-clockwise

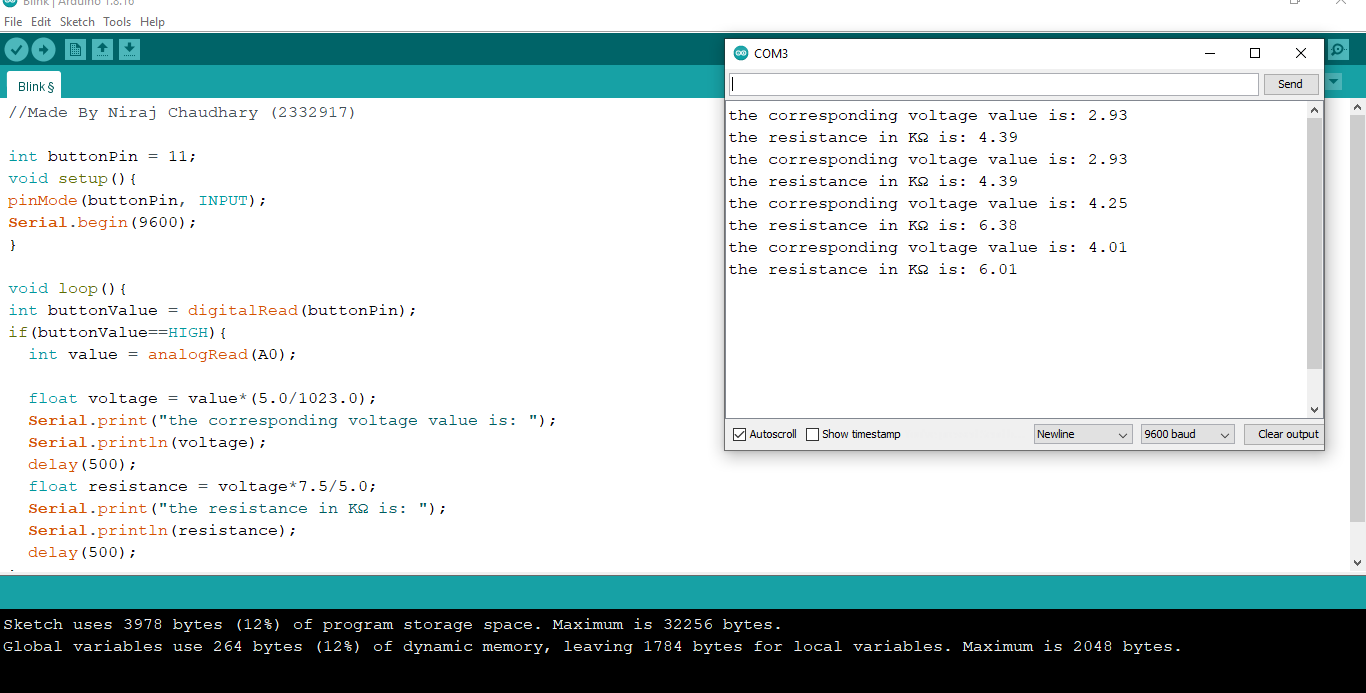
6.85 KΩ

Sample of Values

Pot Resistance against Voltage change

|  |  |
| --- | --- |
| Pot Resitance | Voltage Measured |
| 7.03 KΩ | 4.69 V |
| 6.59 KΩ | 4.39 V |
| 6.28 KΩ | 4.19 V |
| 5.85 KΩ | 3.90 V |
| 3.96 KΩ | 2.64 V |

Screen Shot of Meaningful Serial Port Output, not just numbers



## Activity 4.4: Temperature Sensor & Serial Port

Code - Centigrade to Serial port, but when button Pressed Fahrenheit Displayed Instead

//Made By Niraj Chaudhary (2332917)

#include <dht.h>

#define dataPin 8

dht DHT;

int buttonPin = 11;

void setup()

{

Serial.begin(9600);

}

void loop()

{

int readData = DHT.read11(dataPin);

float t = DHT.temperature;

float h = DHT.humidity;

if (digitalRead(buttonPin) == HIGH){

Serial.print("Temperature = ");

Serial.print((t \* 9.0) / 5.0 + 32.0);

Serial.print(" ");

Serial.print("F");

}

else

{

Serial.print("Temperature = ");

Serial.print(t);

Serial.print(" ");

Serial.print("C");

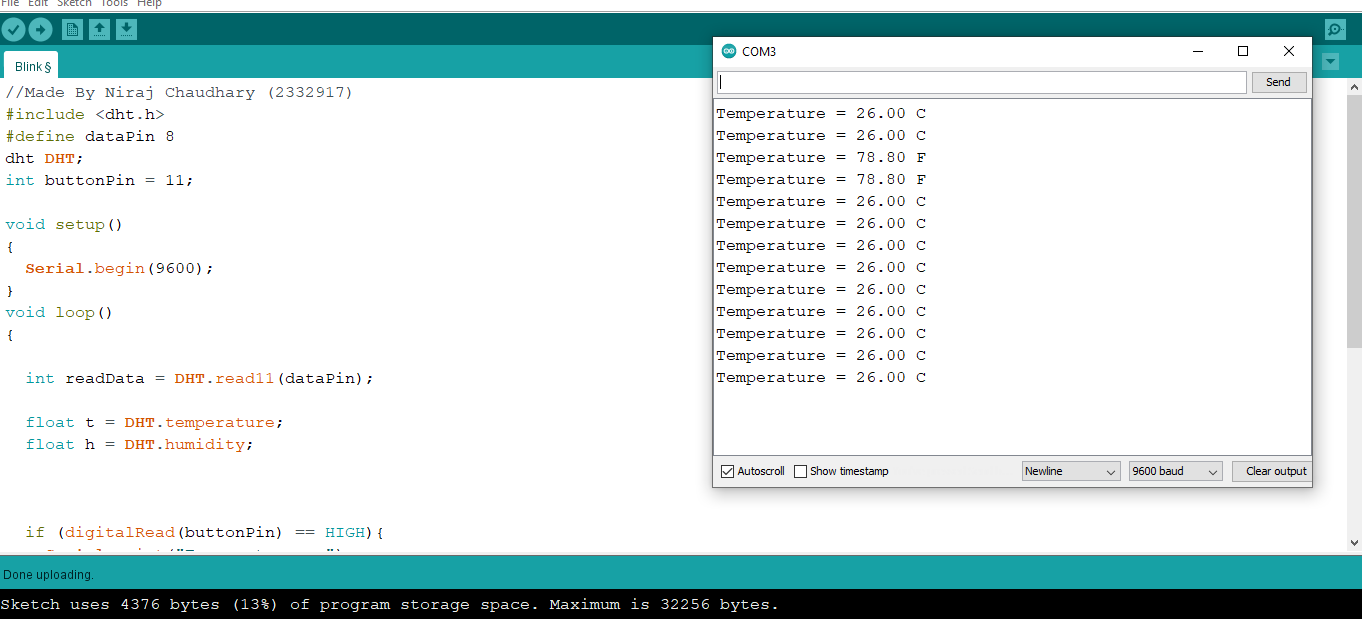
}

Serial.println("");

delay(2000);

}

Screen Shot of Serial Port

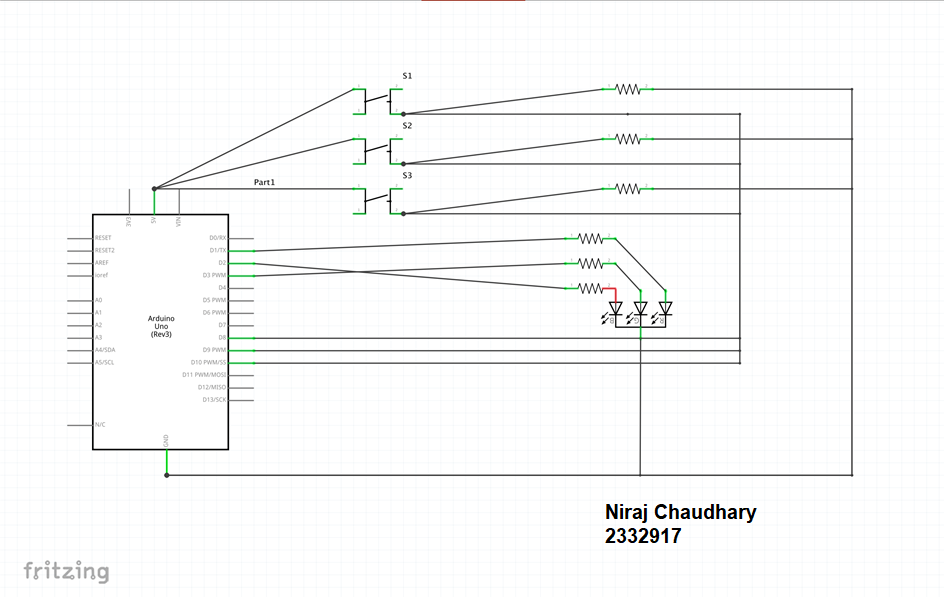


# 

# Workbook 5

## Activity 5.1: RGB Led and switches

Fritzing Arduino Program



**Arduino Program**

// Niraj Chaudhary

// ( 2332917 )

int pinOne = 1;

int pinTwo = 2;

int pinThree = 3;

int buttonPin1 = 8;

int buttonPin2 = 9;

int buttonPin3 = 10;

void setup()

{

pinMode(pinOne, OUTPUT);

pinMode(pinTwo, OUTPUT);

pinMode(pinThree, OUTPUT);

pinMode(buttonPin1, OUTPUT);

pinMode(buttonPin2, OUTPUT);

pinMode(buttonPin3, OUTPUT);

}

void loop()

{

if (digitalRead(buttonPin1) == 1){

digitalWrite(pinOne, 1);

digitalWrite(pinTwo, 0);

digitalWrite(pinTwo, 0);

}

if (digitalRead(buttonPin2) == 1){

digitalWrite(pinOne, 0);

digitalWrite(pinThree, 0);

digitalWrite(pinTwo, 1);

}

if (digitalRead(buttonPin3) == 1){

digitalWrite(pinOne, 0);

digitalWrite(pinTwo, 0);

digitalWrite(pinThree, 1);

}

}

## Activity 5.2: Distance Sensor

**Arduino Code**

//Niraj Chaudhay

// 2332917

int echoPin = 3;

int trigPin = 2;

int pin2 = 7;

void setup()

{

pinMode(trigPin, OUTPUT);

pinMode(pin2,OUTPUT);

pinMode(echoPin, INPUT);

Serial.begin(9600);

}

void loop()

{

long highPulseDuration;

int calculatedDistanceCm;

digitalWrite(trigPin, 0);

delayMicroseconds(5);

digitalWrite(trigPin,1);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

highPulseDuration = pulseIn(echoPin, HIGH);

calculatedDistanceCm = highPulseDuration \*0.034 / 2;

Serial.print("Calculated Distance: ");

Serial.print(calculatedDistanceCm);

Serial.println(" cm");

if (calculatedDistanceCm < 40 ){

digitalWrite(pin2,1);

}

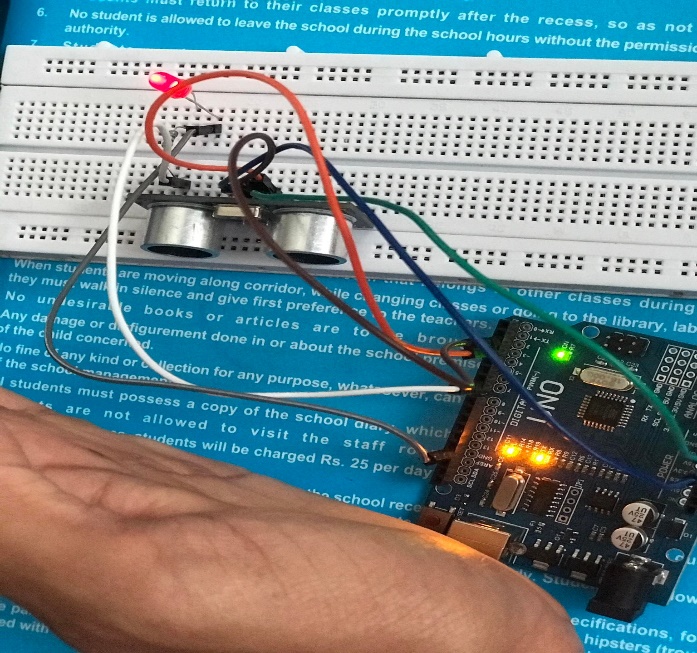
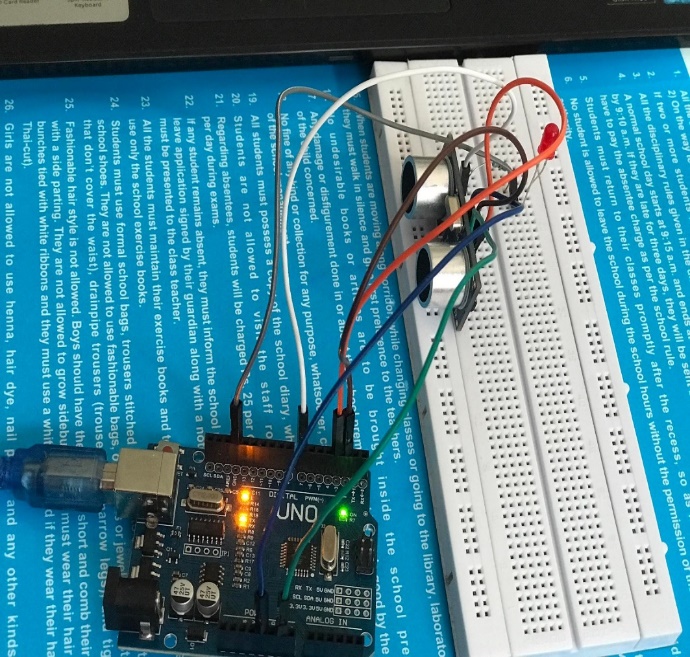
else{

digitalWrite(pin2,0);

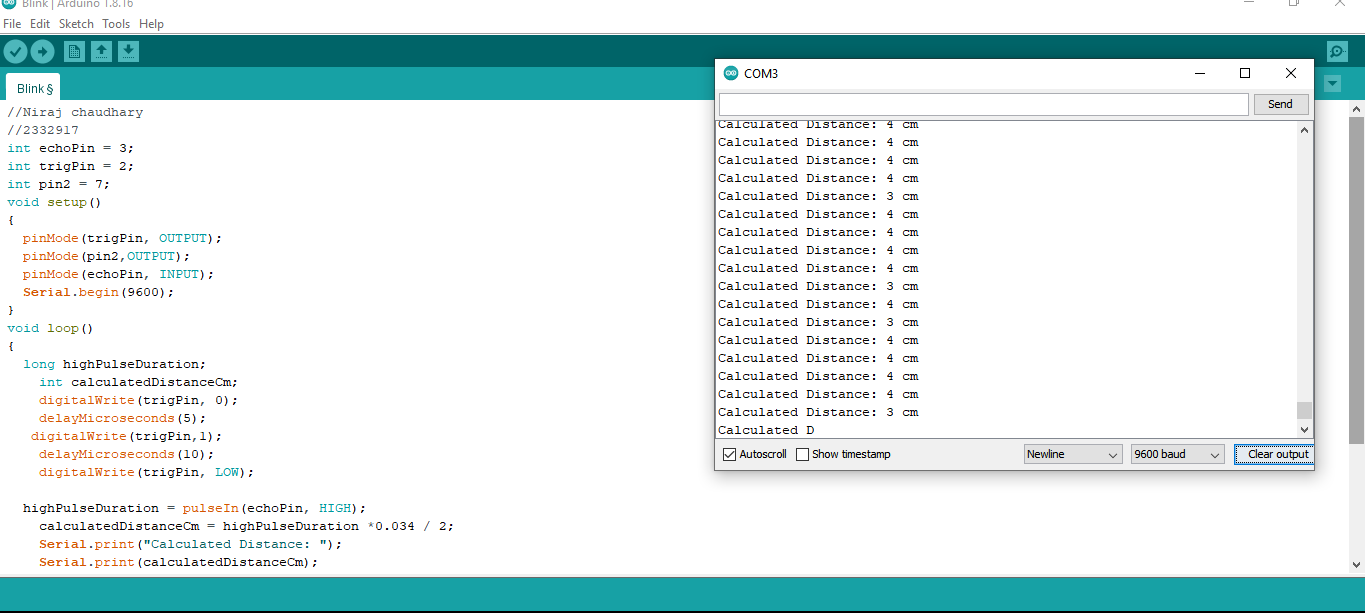
}

};

Take a picture of your distance sensor and include it here, please reduce the size and quality as it will be too large else ☺

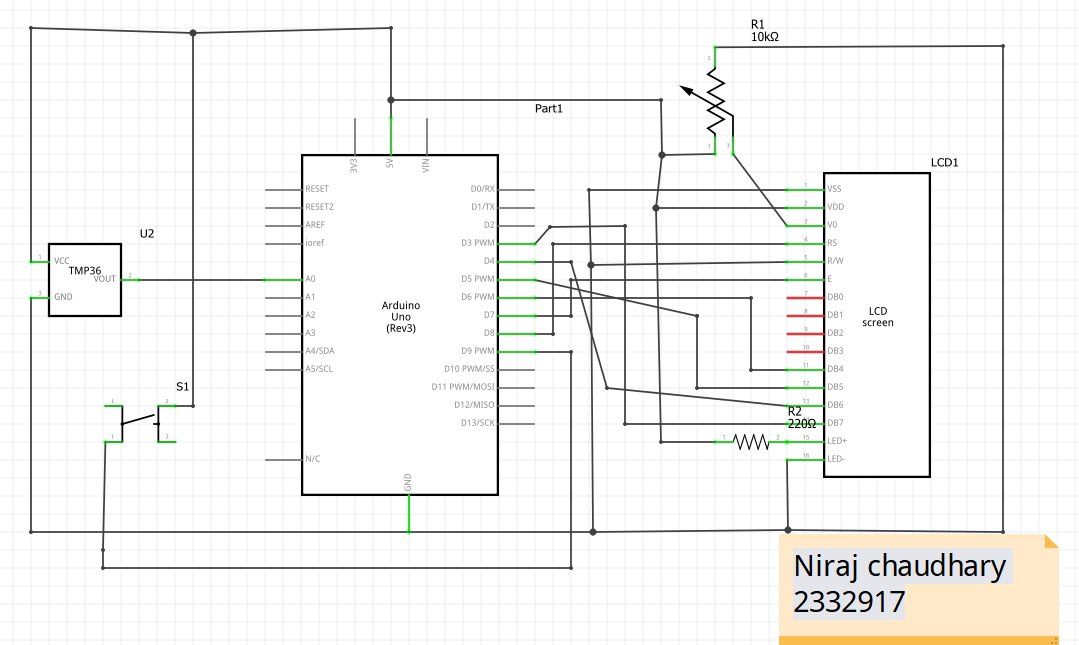


Output



## Activity 5.3: 1602 LCD Display

Fritzing



**Arduino Program**

// Niraj chaudhary

// 2332917

#include "LiquidCrystal.h"

#include <dht.h>;

LiquidCrystal lcd(8,7,6,5,4,3);

int dataPin = 12;

dht DHT;

int buttonPin = 13;

void setup()

{

Serial.begin(9600);

lcd.begin(16,2);

pinMode(buttonPin,OUTPUT);

}

void loop()

{

int readData = DHT.read11(dataPin); // DHT11

float t = DHT.temperature;

float h = DHT.humidity;

if (digitalRead(buttonPin) == HIGH){

lcd.setCursor(0,0);

lcd.print("Temperature");

lcd.setCursor(8,1);

lcd.print(" fahren");

lcd.setCursor(0,1);

lcd.print((t \* 9.0) / 5.0 + 32.0);

}

else

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Temperature = ");

lcd.setCursor(0,1);

lcd.print(t);

lcd.setCursor(8,1);

lcd.print("degC");

}

Serial.println("");

delay(1000);

}

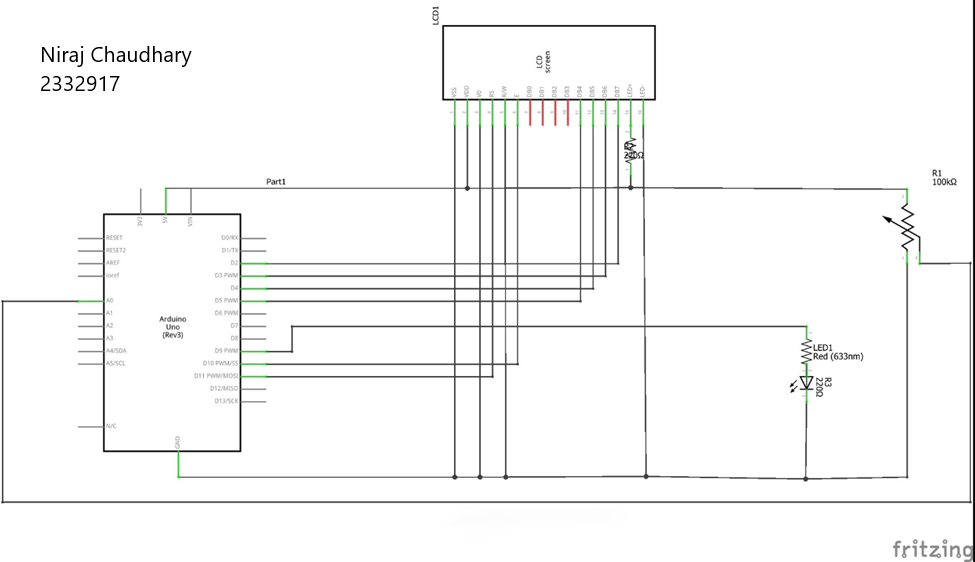
Take a picture of your LCD and include it here, please reduce the size and quality as it will be too large else ☺



# Workbook 6

## Activity 6.1: PWM

Fritzing



**Arduino Program**

//Niraj Chaudhary

// 2332917

#include<LiquidCrystal.h>

int sensorValue = 0; // intitalize the library with the numbers of the interface pin LiquidCrystal lcd(11, 10, 5, 4, 3, 2);

  void setup() {   // declaring pinmode and displaying initial Message in LCD

  pinMode(A0, INPUT);

  pinMode(9, OUTPUT);

  Serial.begin(9600);

  lcd.begin(16, 2);

  lcd.print("Starting System");

delay(1000);

  lcd.clear();

lcd.print("System On");

delay(1000);

lcd.clear(); }

  void loop() {

lcd.setCursor(0, 1); // setting the cursor to the second row

  lcd.print("Potentiometer:");

sensorValue = analogRead(A0);

  lcd.print(sensorValue);

int brightness = sensorValue;

  analogWrite(9, brightness); // setting the brightness

  Serial.println(sensorValue);

  lcd.setCursor(0, 0); // setting the cursor to first row

lcd.setCursor(4, 1);

  lcd.print("ADC Value"); // display value

delay(500);

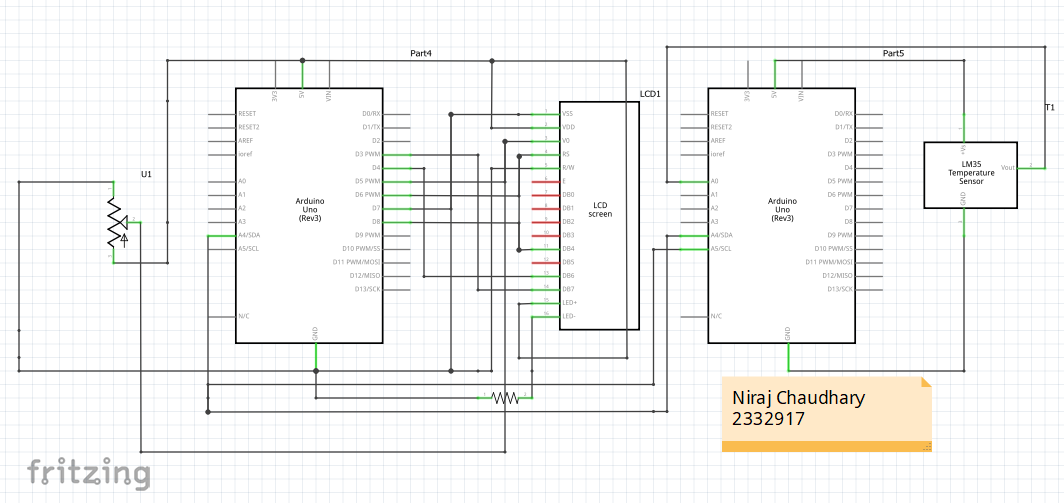
lcd.clear();

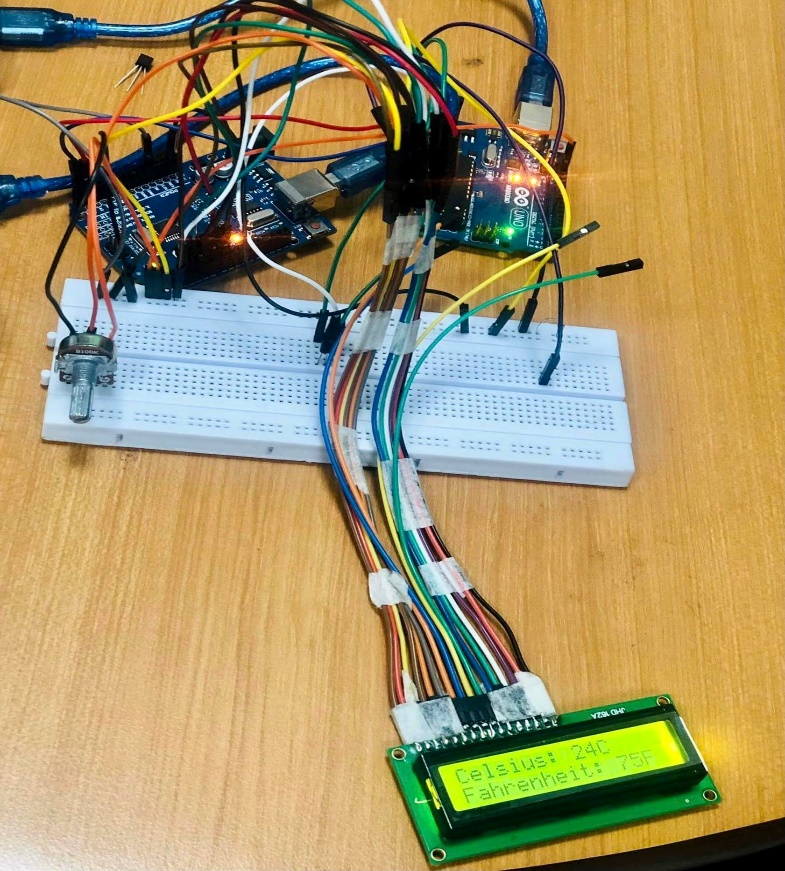
}

# Workbook 7

## Activity 7.1: 2 Arduinos – using Digital Pins

Fritzing





**Arduino Program**

//Niraj Chaudhary

//2332917

//Code1

#include<LiquidCrystal.h>

#include <Wire.h>

LiquidCrystal lcd(8, 7, 6,5,4,3);

 void setup() {

 lcd.begin(16,2);

 Serial.begin(9600);

 Wire.begin(8);

 lcd.print("Sending data...");

lcd.setCursor(0,1);

  lcd.print("Please wait...");

 pinMode(8, OUTPUT); }

void loop() {

  Wire.onReceive(dataReceived);

}

void dataReceived(int x) //x means how many times data received

{

 int CelsiusVal = Wire.read();

int FahrenheitVal = Wire.read();

  lcd.clear();

 lcd.setCursor(0, 0);

 lcd.print("Celsius: ");

lcd.print(CelsiusVal);

lcd.print("C");

     lcd.setCursor(0, 1);

lcd.print("Fahrenheit: ");

  lcd.print(FahrenheitVal);

 lcd.print("F");

    }

//Code 2

#include <Wire.h> int sensorPin = 0;

void setup()

{

 Serial.begin(9600);

 Wire.begin(); }

void loop()

    {

   int reading = analogRead(sensorPin);

   float voltage = reading \* 5.0;

 voltage /= 1024.0;

 int temperatureC = (voltage - 0.5) \* 100 ;

Serial.print(temperatureC);

Serial.println(" degrees C");

 Wire.beginTransmission(8);

 Wire.write(temperatureC);

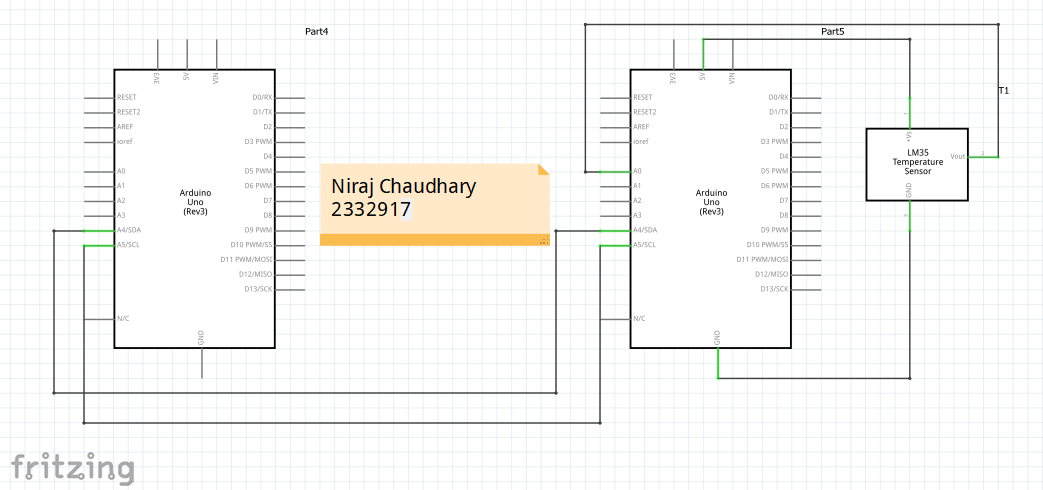
Wire.endTransmission();

delay(1000);

}

## Activity 7.2: 2 Arduinos – using Serial I/O

Fritzing



**Arduino Program**

//Niraj Chaudhary

//2332917

//Code 1

#include <Wire.h>

int sensorPin = 0;

void setup()

{

 Serial.begin(9600);

  Wire.begin();

}

void loop()

  {

   int reading = analogRead(sensorPin);

   float voltage = reading \* 5.0;

 voltage /= 1024.0;

   int temperatureC = (voltage - 0.5) \* 100 ;

   Serial.print(temperatureC);

Serial.println(" degrees C");

 Wire.beginTransmission(8);

 Wire.write(temperatureC);

Wire.endTransmission();

   delay(1000);

}

//Code2

#include <LiquidCrystal.h>

#include <Wire.h>

LiquidCrystal lcd(8, 7, 6,5,4,3);

void setup() {

lcd.begin(16,2);

  Serial.begin(9600);

Wire.begin(8);

serial.print("Sending data...");

 lcd.setCursor(0,1);

  serial.print("Please wait...");

  pinMode(8, OUTPUT);

}

  void loop() {

  Wire.onReceive(dataReceived);

}

  void dataReceived(int x)//x means how many times data received

{

    int CelsiusVal = Wire.read();

  int FahrenheitVal = Wire.read();

  Serial.clear();

 Serial.setCursor(0, 0);

  Serial.print("Celsius: ");

  Serial.print(CelsiusVal);

 Serial.print("C");

    Serial.setCursor(0, 1);

 Serial.print("Fahrenheit: ");

Serial.print(FahrenheitVal);

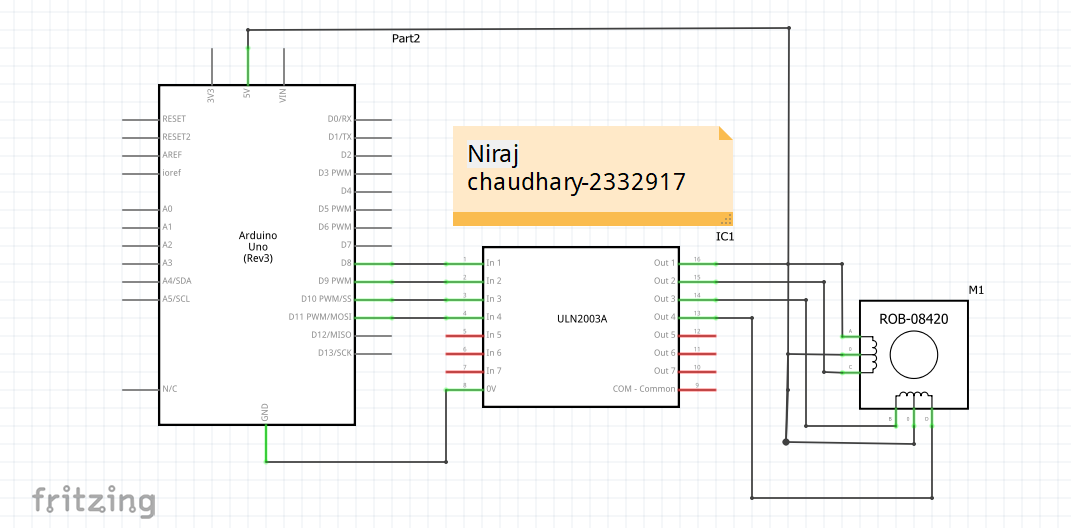
 Serial.print("F");

  }

# Workbook 8

## Activity 8.1: Stepper Motor Circuit Diagram

Circuit Diagram



**Arduino Program**

//Niraj Chaudhary

//2332917

## //Includes the Arduino Stepper Library

## #include <Stepper.h>

## // Defines the number of steps per rotation const int stepsPerRevolution = 2038; // Creates an instance of stepper class

## // Pins entered in sequence IN1-IN3-IN2-IN4 for proper step sequence Stepper myStepper = Stepper(stepsPerRevolution, 8, 10, 9, 11); void setup()

## {

## // Nothing to do (Stepper Library sets pins as outputs) }

## void loop()

## {

## // Rotate CW slowly at 5 RPM

## myStepper.setSpeed(5);

## myStepper.step(stepsPerRevolution);

## delay(1000);

## // Rotate CCW quickly at 10 RPM

## myStepper.setSpeed(10);

## myStepper.step(-stepsPerRevolution);

## delay(1000);

## }

## Activity 8.2: 2 Stepper Motors

**Arduino Program**

//Niraj Chaudhary - 2332917

// Include the AccelStepper library:

#include <AccelStepper.h>

// Motor pin definitions:

#define motorPin1 8 // IN1 on the ULN2003

driver #define motorPin2 9 // IN2 on the ULN2003

driver #define motorPin3 10 // IN3 on the ULN2003

driver #define motorPin4 11 // IN4 on the ULN2003

driver #define motorPin5 3

#define motorPin6 4

#define motorPin7 5

#define motorPin8 6

// Define the AccelStepper interface type; 4 wire motor in half step mode:

#define MotorInterfaceType 8 // Initialize with pin sequence IN1-IN3-IN2-IN4 for using the AccelStepper library with 28BYJ-48 stepper motor:

AccelStepper stepper1 = AccelStepper(MotorInterfaceType, motorPin1, motorPin3, motorPin2, motorPin4);

AccelStepper stepper2= AccelStepper(MotorInterfaceType, motorPin5, motorPin6, motorPin7, motorPin8);

void setup()

{ // Set the maximum steps per second:

stepper1.setMaxSpeed(1000);

stepper2.setMaxSpeed(1000);

}

void loop()

{ // Set the current position to 0:

stepper1.setCurrentPosition(0);

stepper2.setCurrentPosition(0);

//Forward Rotation until one revolution while (stepper1.currentPosition() != 2038) { stepper1.setSpeed(500);

stepper1.runSpeed();

}

while (stepper2.currentPosition() != 2038)

{

stepper2.setSpeed(500);

stepper2.runSpeed();

}

delay(1000);

// Reset the position to 0:

stepper1.setCurrentPosition(0);

stepper2.setCurrentPosition(0); //Backward rotation Until One Revolution

while (stepper1.currentPosition() != -2038) {

stepper1.setSpeed(500);

stepper1.runSpeed();

}

while (stepper2.currentPosition() != -2038) {

stepper2.setSpeed(500);

stepper2.runSpeed();

}

delay(1000);

// Reset the position to 0:

stepper1.setCurrentPosition(0);

stepper2.setCurrentPosition(0);

//Stepper 1 Forward, Stepper 2 Reverse rotation

while (stepper1.currentPosition() != 2038)

{

stepper1.setSpeed(500);

stepper1.runSpeed();

} while (stepper2.currentPosition() != -2038) {

stepper2.setSpeed(500);

stepper2.runSpeed();

}

delay(1000); // Reset the position to 0:

stepper1.setCurrentPosition(0);

stepper2.setCurrentPosition(0);

//Stepper 1 Forward, Stepper 2 Reverse rotation

while (stepper1.currentPosition() != -2038)

{

stepper1.setSpeed(500);

stepper1.runSpeed();

}

while (stepper2.currentPosition() != 2038)

{

stepper2.setSpeed(500);

stepper2.runSpeed();

}

delay(1000);

}

# Workbook 9

## Activity 9.1: Windscreen Wiper Code using Servos & Temperature Sensor

**Arduino Code**

//Niraj Chaudhary - 2332917

#include <Servo.h>

Servo myservo;

int servoPin = ;

int sensorPin = 0;

void setup() {

  Serial.begin(9600);

myservo.attach(servoPin);

}

  void loop()

  {

 int reading = analogRead(sensorPin);

   float voltage = reading \* 5.0;

 voltage /= 1024.0;

   Serial.print(voltage);

Serial.println(" volts");

 float temperatureC = (voltage) \* 100 ;

 Serial.print(temperatureC);

Serial.println(" degrees C");

delay(1000);

 if (temperatureC > 0)

 {

 myservo.write(90);

  delay(1000);

  myservo.write(0);

  delay(1000);

    }

}

# Individual Project (50%)

## Rationale

Throughout the module you have used a range of sensors and actuators with an Arduino to complete weekly tasks. For the mini project we would like you to research and create a small embedded project in an area of your choice, such as:

* Games
* Networking
* IT Security
* Systems Engineering
* Smart Technology
* Artificial Intelligence

Previous projects have included a reaction game that gives a score depending on how fast you hit a button, this has buttons to restart the application, and an LCD to show scores, and information.

This project should be your own work, YOU MUST NOT COPY A PROJECT FROM THE INTERNET.

## Timescales

This project should be started around week 5 and continue until the deadline, when it will be submitted in the Portfolio.

## Equipment

You are free to use Tinkercad, or your own kit.

## The Project

### Step 1 produce adetailed description of your project.

This should clearly describe what you are intending to build and may contain some diagrams of how the sensor/switches input is to be processed by the Arduino. Then what kind of output is intended to be seen or heard by the user. Please mention any tools you intend to use.

### Step 2 Circuit Diagram&Fritzing Schematic

You are required to produce a circuit diagram of your work showing any calculations you made, so these might be suitable resistor values for any LED’s you use. These calculations are covered on the module. The circuit diagram should not be hand drawn but should follow the format of circuits from the module.

### Step 3 A Program

You will need to write some software for this project and a listing of the code with suitable comments will need to be included.

### Step 4 Testing

You will be required to produce some suitable test data that you would expect to be able to measure such as voltages, test code.

Once your prototype is complete you will be expected to test your circuit and compare the actual values to your initial test data, and comment on the results.

### Step 5 Conclusions

You are required to write a summary of the work along with a short half page reflection on how you found the work.

### Layout

The report should be suitably laid out for a report, using headings, references if required in Harvard style, and appendices used for any lengthy code. All diagrams should be produced on a PC, and hand-written work is not acceptable.

### Marking

# All sections carry equal marks.