Student Name
Nayan Raj Khanal

Student Number
2227486

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

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.

5V

Activity 1.2: Actual voltage across 3.3V breadboard pins.

3.3V

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.

There is no tolerance to affect the circuit because TinkerCad is not impacted by any external elements like heat, temperature, internal resistance, etc. However, due to internal wire resistance in the multimeter, this is not possible. Additionally, temperature and heat play important roles in developing tolerance. This explains why values in Tinkercad and values obtained from a multimeter are different.

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

Actual value in Circuit (a) =4.84V

Labelled value (l) =5v

Tolerance = (l-a)/l*100

=3.2%

Activity 1.3: Potential Divider Calculations

Show the working on how you achieved 2.5V

2.5v can be achieved by using two resistors of same resistance and connect them in the series as resistors when connected in series act as potential divider. As we have used resistors of same resistance, the voltage is divided equally and we can get 2.5v

In calculation;

We have:

Volt=5v

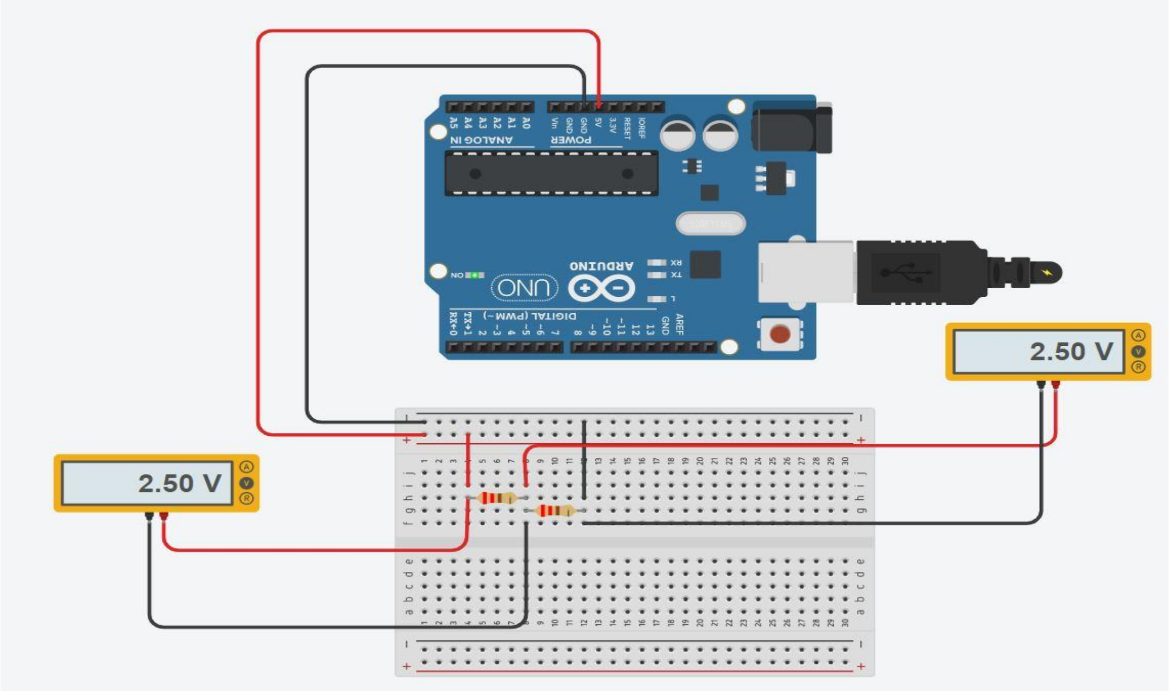
Resistance in one volt=1=220ohm

Total resistance=220+220=440ohm

V after=(R/R total)*V

=(220/440)*5

V=2.5v



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

Since the potential drop is 5v and we have 3v drop through one resistor and so simply the other must have 2v drop

Voltage (V1) =3v

Resistance=220ohm

Current (I) =?

We have,

V1=IR

I=v1/r

So I=3/220

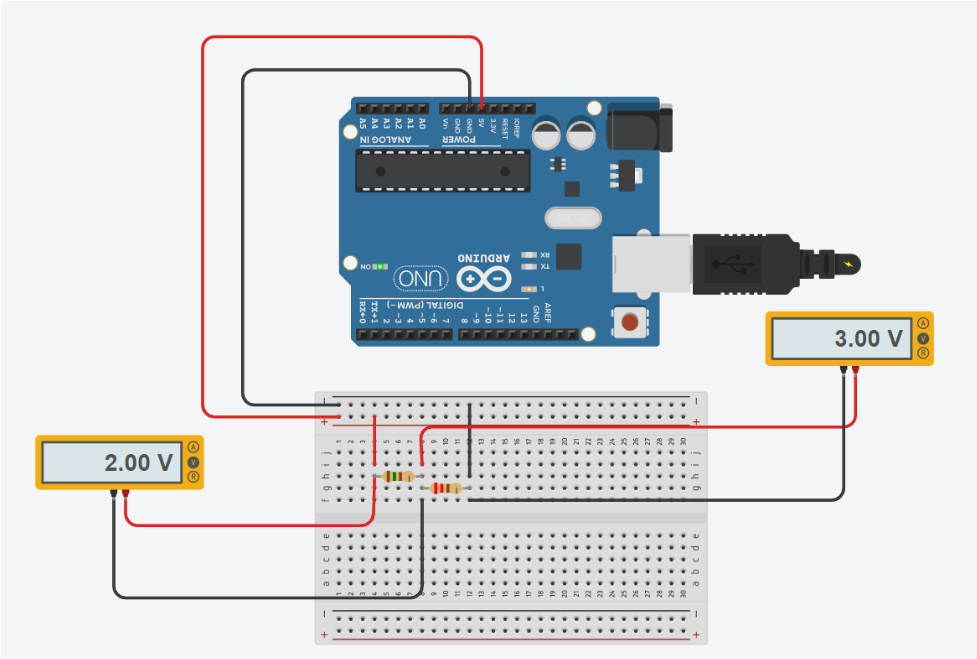
Since, the current flowing through both the resistor would be same;

V2=IR2

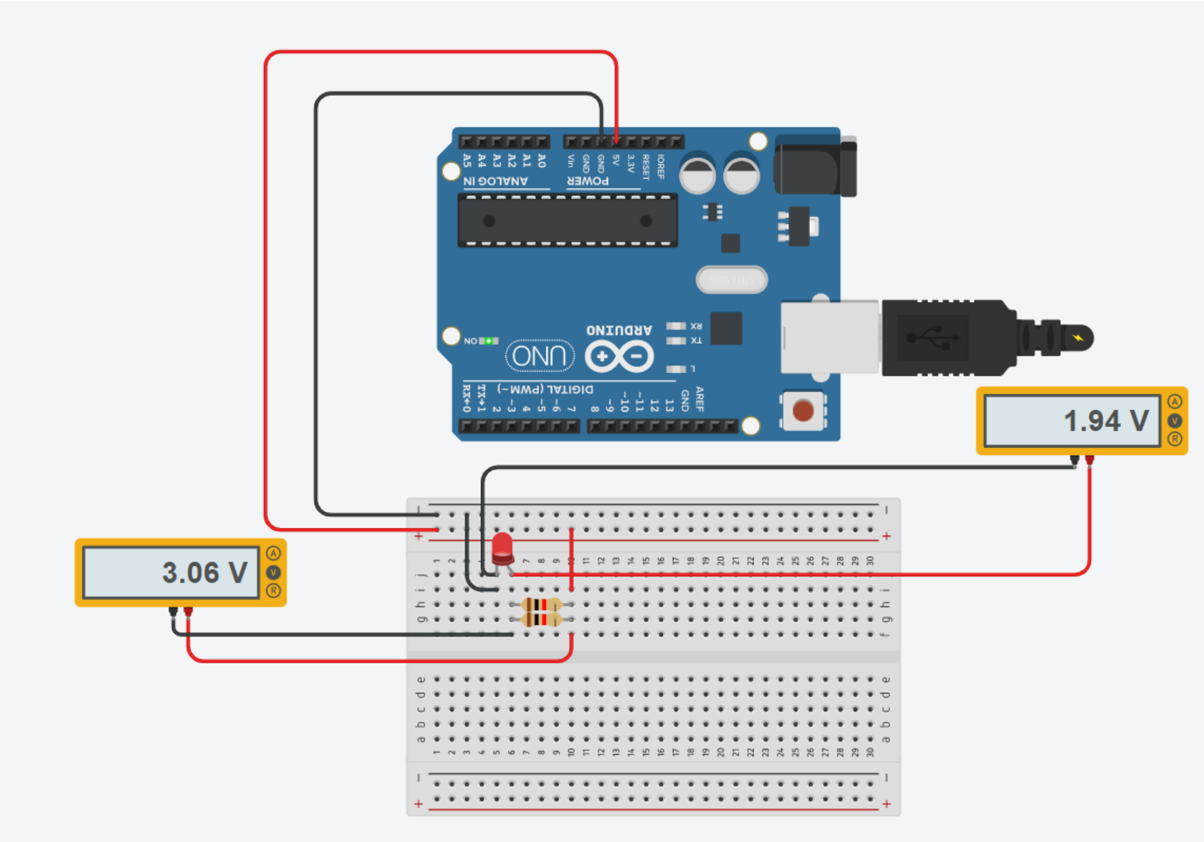
2= (3/220) *R2

R2=146.67

So we can use one resistor of resistance 220ohm and another of approximately 146.47ohm to get the 3v and 2v drop through R1 and R2 respectively.



Activity 1.5: Voltage Divider circuit readings from Breadboard circuit.



Activity 1.6: LED Circuits

Each resistor Value

220 ohm

220 ohm

Total resistance Calculation

Here, R1=220 ohm, and R2=220 ohm.

Now,

$1/R=1/R1 +1/R2$

$1/R = 1/220 + 1/220$

$1/R=2/220$

R=110ohm

Measured Resistance

R=109ohm

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

Temperature is influenced by the resistance, which makes the resistance tolerant. We can assert that the simulation depicts the actual value even though losses happen in practice and cause the value to deviate from the original value because no external or internal losses are noticed.

Activity 1.7: Current Measurement

Calculation of current flowing into LED

45 mA

Actual measured value of current

45 mA

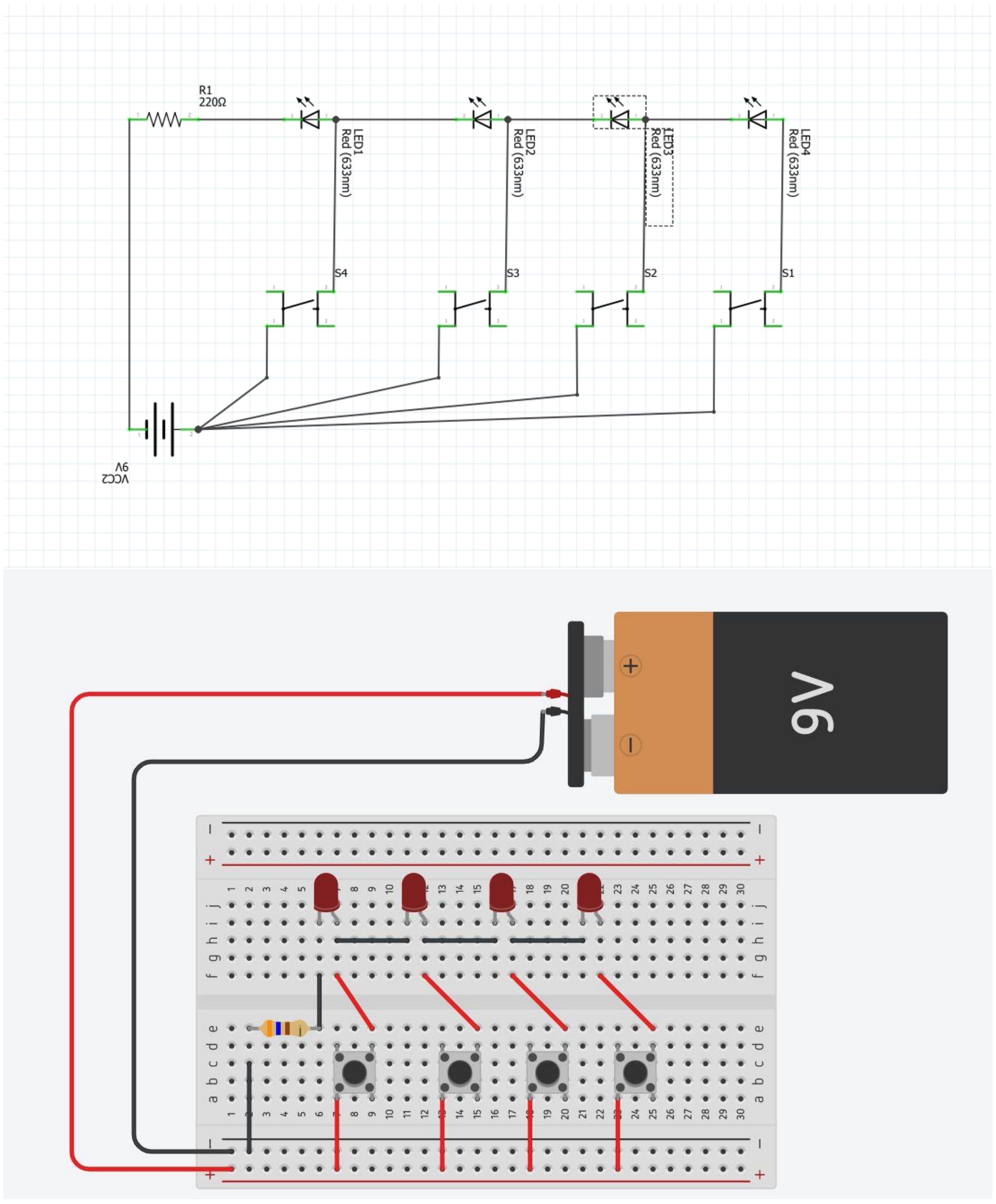
Why might they be different?

Known to us

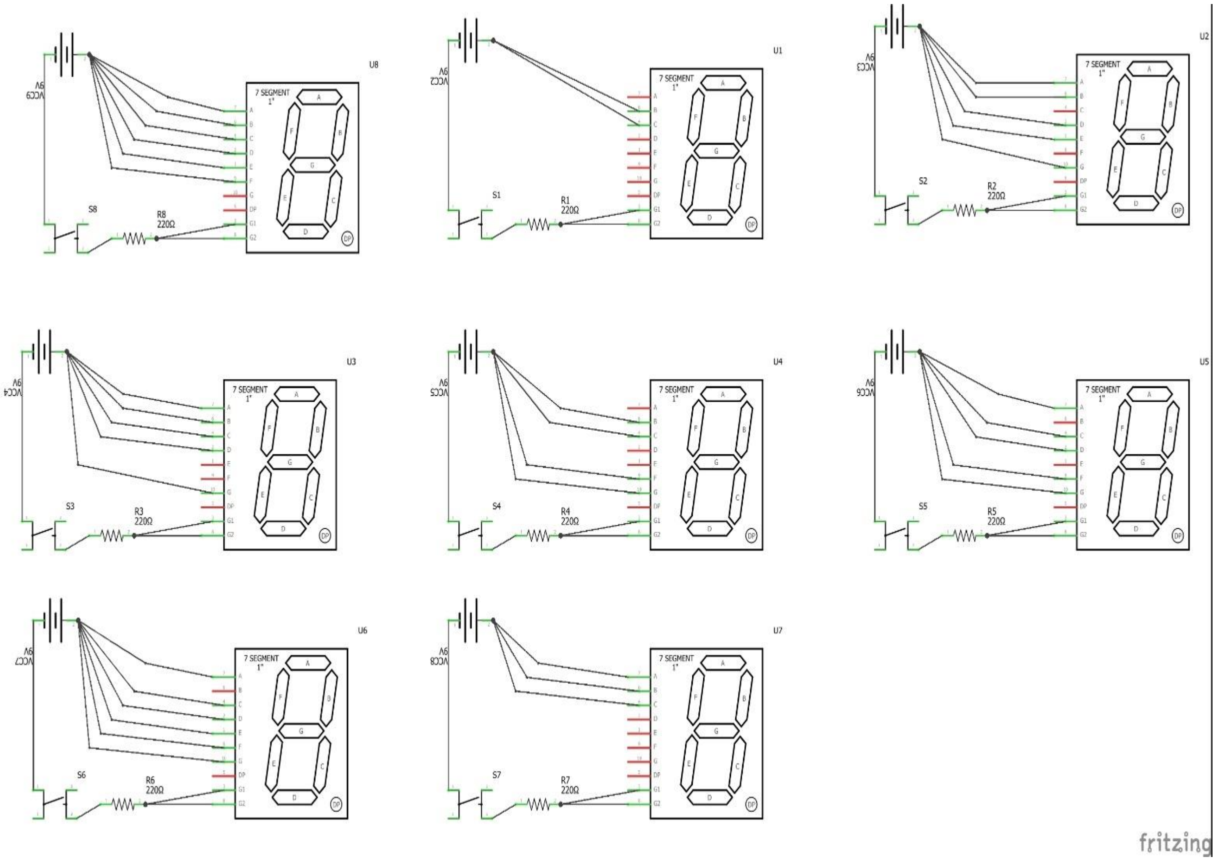
V=IR

Therefore, resistivity may change as a result of internal resistance when we measure the current, changing the value of the current as well.

Activity 1.8: Fritzing for 4 switches & LEDS



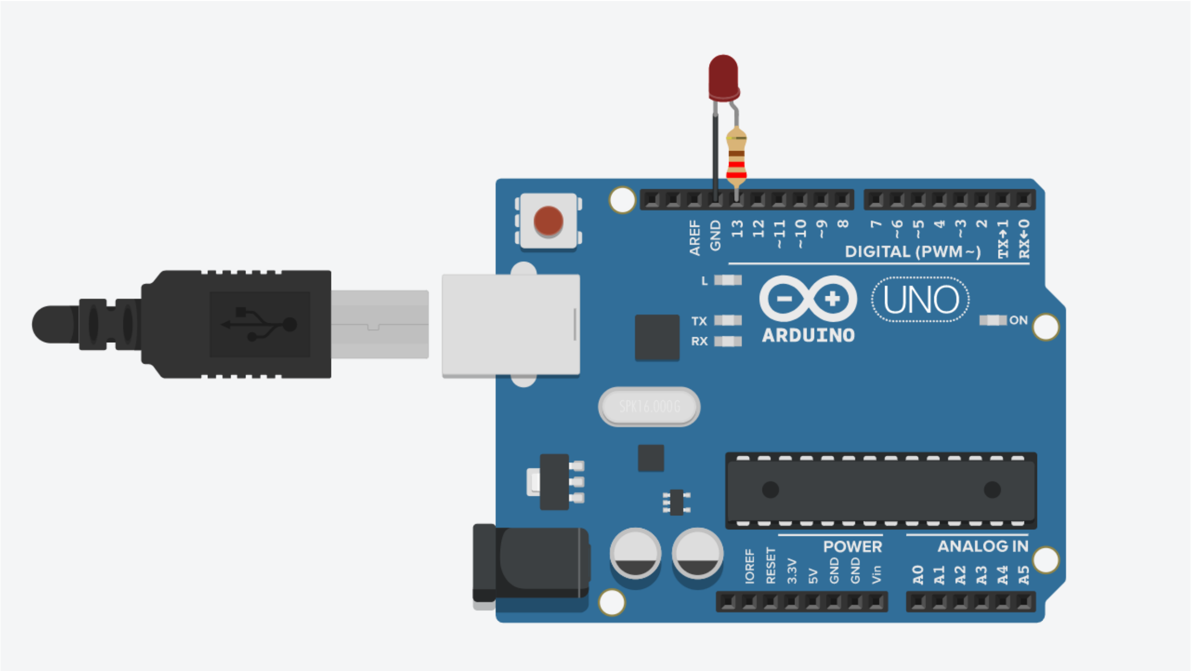
Activity 1.9: Fritzing for Number 0-7



Workbook 2

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

63 as binary, including working



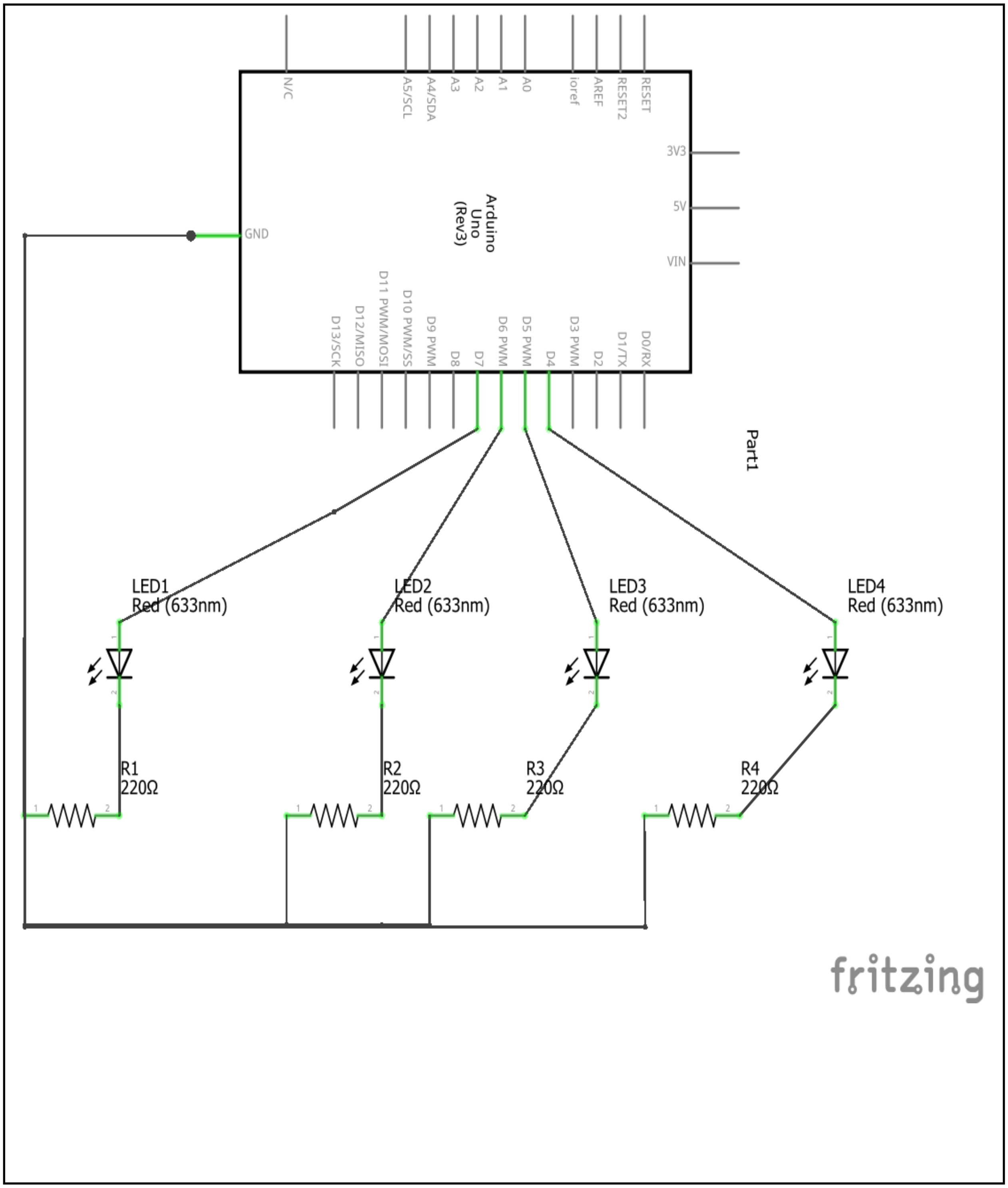
Copy & Post your code with a suitable comment at the top of code with your name & student number

```
/*Binary code for 63 is 111111, meaning the led must be high six times in a row.

(Divide 63 by 2 and write the remainder from bottom to top)

Using one LED and one resistor of resistance 220 ohm

Nayan Raj Khanal

2227486

*/

void setup()

{

pinMode(LED_BUILTIN, OUTPUT); // Initializing pin 13 as output pin

}


void loop()

{

 for(int i=0;i<6;i++) //blink LED 6 times

{

 digitalWrite(LED_BUILTIN, HIGH); //Turning ON LED at Pin 13

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

 digitalWrite(LED_BUILTIN, LOW); //Turning OFF LED at Pin 13

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

 }

 delay(5000); // Wait for 5000 millisecond(s)

}
```

☺

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

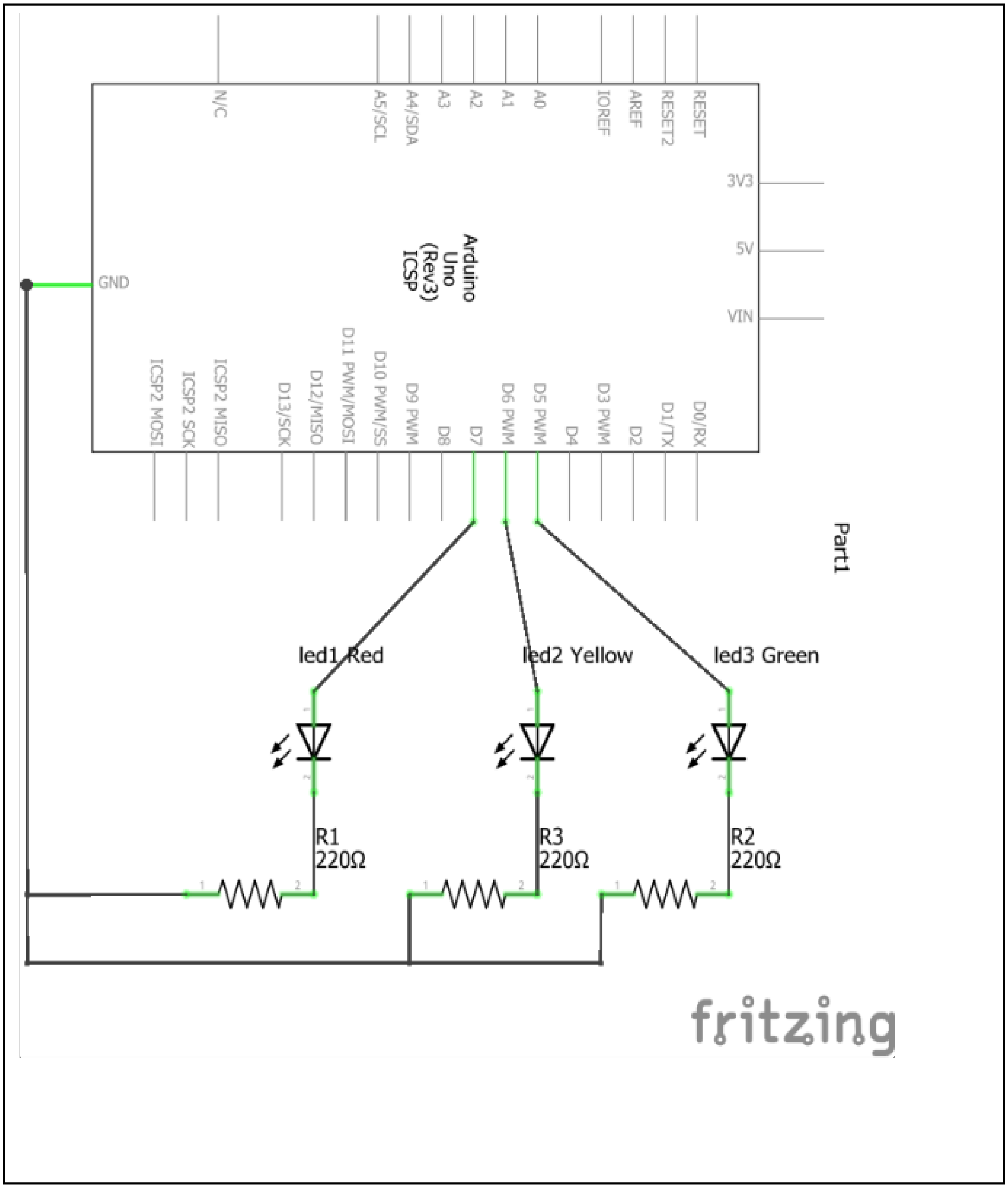


```
// C++ code
//Nayan Raj Khanal
//2227486

int ledOne = 13;
int ledTwo = 12;
int ledThree = 11;
int ledFour = 10;

void setup()
{
  pinMode(ledOne, OUTPUT); // Initializing pin 13 as output pin
  pinMode(ledTwo, OUTPUT); // Initializing pin 12 as output pin
  pinMode(ledThree, OUTPUT); // Initializing pin 11 as output pin
  pinMode(ledFour, OUTPUT); // Initializing pin 10 as output pin

}
void ledOn(int a,int b,int c,int d){
	digitalWrite(ledOne, a);
	digitalWrite(ledTwo, b);
	digitalWrite(ledThree, c);
	digitalWrite(ledFour, d);
	delay(1000);
	digitalWrite(ledOne, 0);
	digitalWrite(ledTwo, 0);
	digitalWrite(ledThree, 0);
	digitalWrite(ledFour, 0);
	delay(1000); // Wait for 1000 millisecond(s)
```

```
void loop()

{

  ledOn(0,0,0,0);

  ledOn(0,0,0,1);

  ledOn(0,0,1,0);

  ledOn(0,0,1,1);

  ledOn(0,1,0,0);

  ledOn(0,1,0,1);

  ledOn(0,1,1,0);

  ledOn(0,1,1,1);

  ledOn(1,0,0,0);

  ledOn(1,0,0,1);

  ledOn(1,0,1,0);

  ledOn(1,0,1,1);

  ledOn(1,1,0,0);

  ledOn(1,1,0,1);

  ledOn(1,1,1,0);

  ledOn(1,1,1,1);

}
```

Activity 2.3: Traffic Lights

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



Arduino Program for Step 4 i.e. 4 LEDs

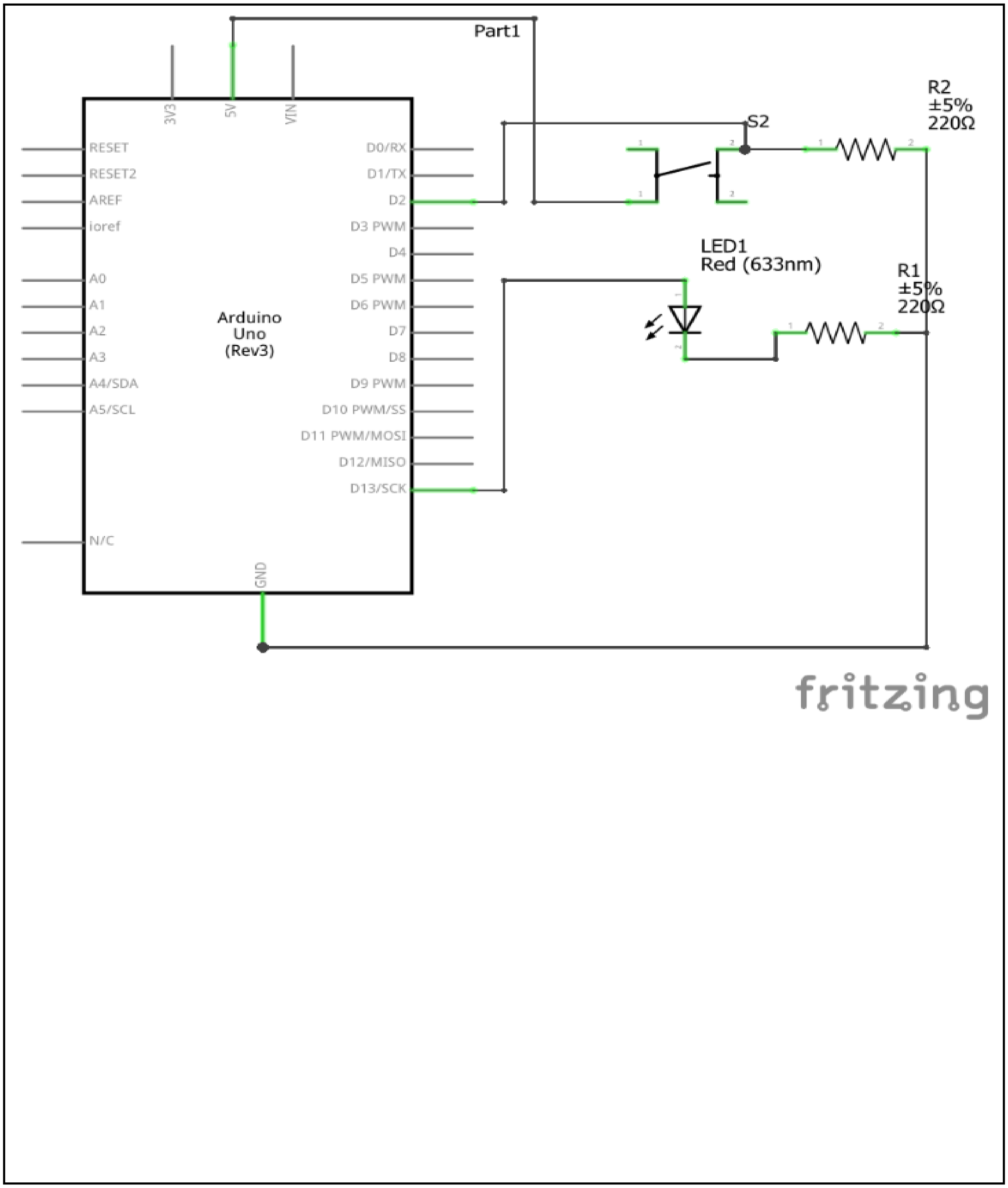
```
//Name:Nayan Raj Khanal
//ID: 2227486

int GREEN = 8;
int YELLOW = 9;
int RED = 10;
int DELAY_GREEN = 5000;
int DELAY_YELLOW = 2000;
int DELAY_RED = 5000;

// basic functions
void setup()
{
  pinMode(GREEN, OUTPUT); // Initializing GREEN as output pin
  pinMode(YELLOW, OUTPUT); // Initializing YELLOW as output pin
  pinMode(RED, OUTPUT); // Initializing RED as output pin
}
void loop()
{
  green_light();
  delay(DELAY_GREEN);
  yellow_light_2();
  delay(DELAY_YELLOW);
  red_light();
  delay(DELAY_RED);
  yellow_light();
  delay(DELAY_YELLOW);
void red_light()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, LOW);
  digitalWrite(RED, HIGH);
}
void green_light()
{
  digitalWrite(GREEN, HIGH);
  digitalWrite(YELLOW, LOW);
  digitalWrite(RED, LOW);
}
yellow_light();
  delay(DELAY_YELLOW);

}

void red_light()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, LOW);
  digitalWrite(RED, HIGH);
}
void green_light()
{
  digitalWrite(GREEN, HIGH);
  digitalWrite(YELLOW, LOW);
  digitalWrite(RED, LOW);
}
```

```
void yellow_light()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(RED, HIGH);
}
void yellow_light_2()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(RED, LOW);
}
void yellow_light()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(RED, HIGH);
}

void yellow_light_2()
{
  digitalWrite(GREEN, LOW);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(RED, LOW);
}
```

Workbook 3

Activity 3.1: Circuit Diagram of Button & LED

Fritzing

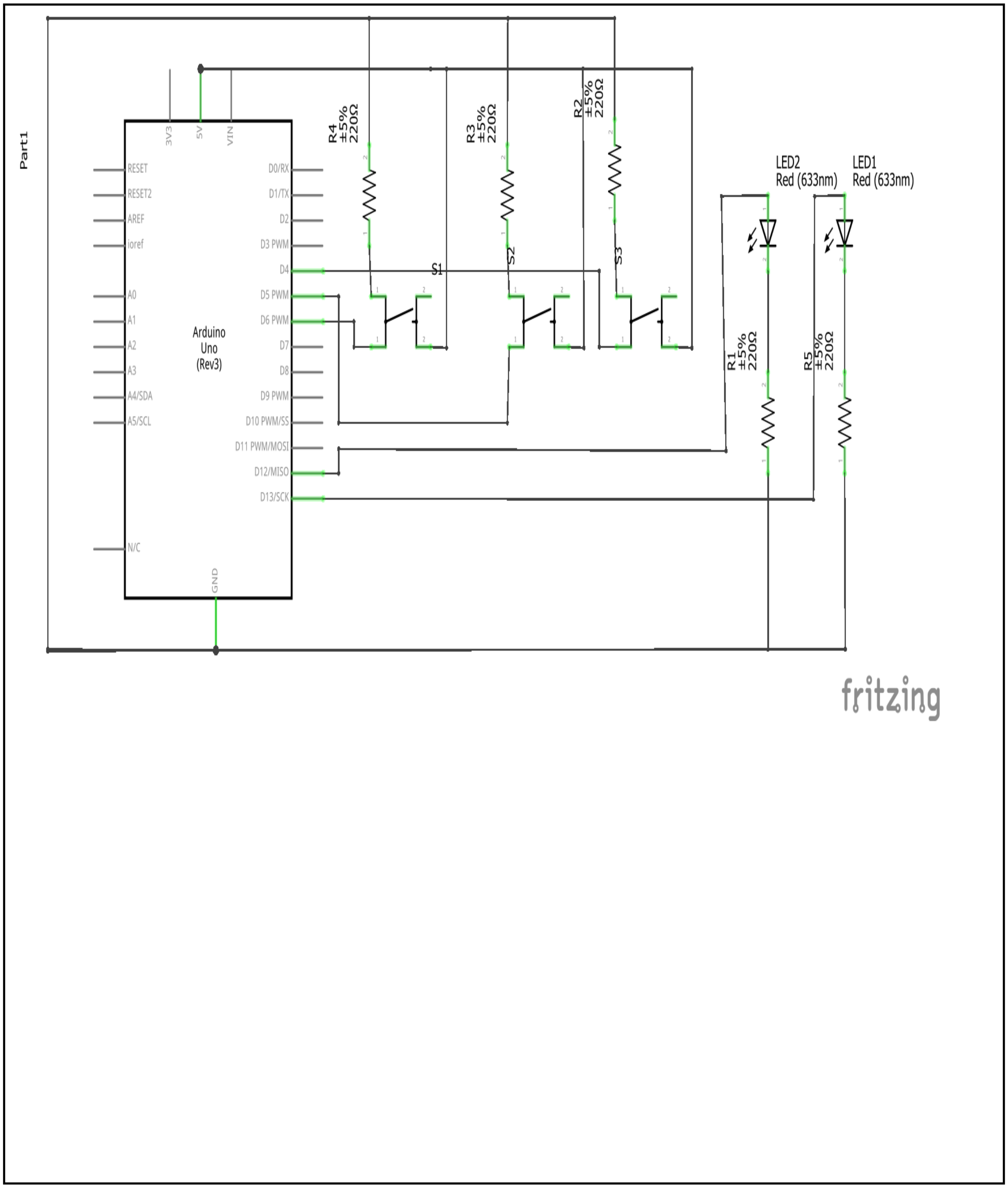


Arduino Program

```
/*
C++ code
Nayan Raj Khanal
2227486
This program employs a pull-up resistor with an extra-large capacity to cut down on self-generated electrical noise.
*/
int buttonPin =2;
int Ledpin=13;
void setup()
{
  pinMode(2, INPUT); // Initializing pin 2 as input pin
  pinMode(13, OUTPUT); // Initializing pin 13 as output pin
}

void loop()
{
  int input=digitalRead(2);
  if (input ==HIGH){
    digitalWrite(13,HIGH);

  }
  else {
    digitalWrite(13,LOW);
}
```

Activity 3.2: 3 Switches & Led

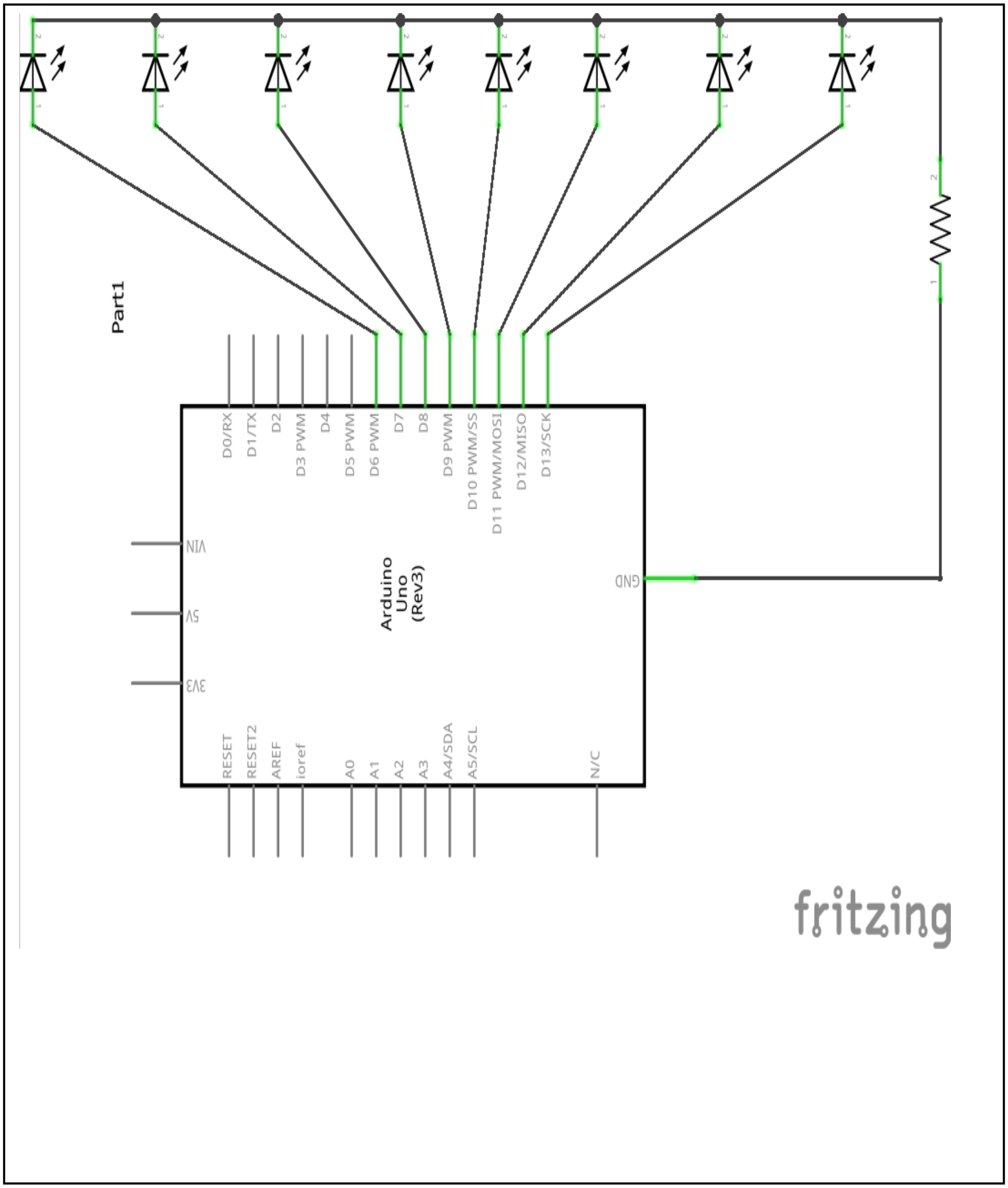
Fritzing Circuit Diagram



```
//Name:Nayan Raj Khanal
//ID: 2227486

int redLed = 13;
int blueLed = 12;
int switch1 = 6;
int switch2 = 5;
int switch3 = 4;

void setup()
{
  pinMode(redLed, OUTPUT); // Initializing redLed as output pin
  pinMode(blueLed, OUTPUT); // Initializing blueLed as output pin
  pinMode(switch1, INPUT); // Initializing switch1 as input pin
  pinMode(switch2, INPUT); // Initializing switch2 as input pin
  pinMode(switch3, INPUT); // Initializing switch3 as input pin

}

void loop()
{

  int switchState1 = digitalRead(switch1);
  int switchState2 = digitalRead(switch2);
  int switchState3 = digitalRead(switch3);

  if(switchState1 == HIGH){
        digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
  }
else if(switchState2 == HIGH){
              digitalWrite(12, HIGH);
    delay(2000);
    digitalWrite(12, LOW);
  }

  else if(switchState3 == HIGH){
              digitalWrite(13, HIGH);
    delay(3000);
    digitalWrite(13, LOW);
  }
}
```

Activity 3.3: 8 Buttons & LEDs (SWITCH STATEMENTS)

Fritzing



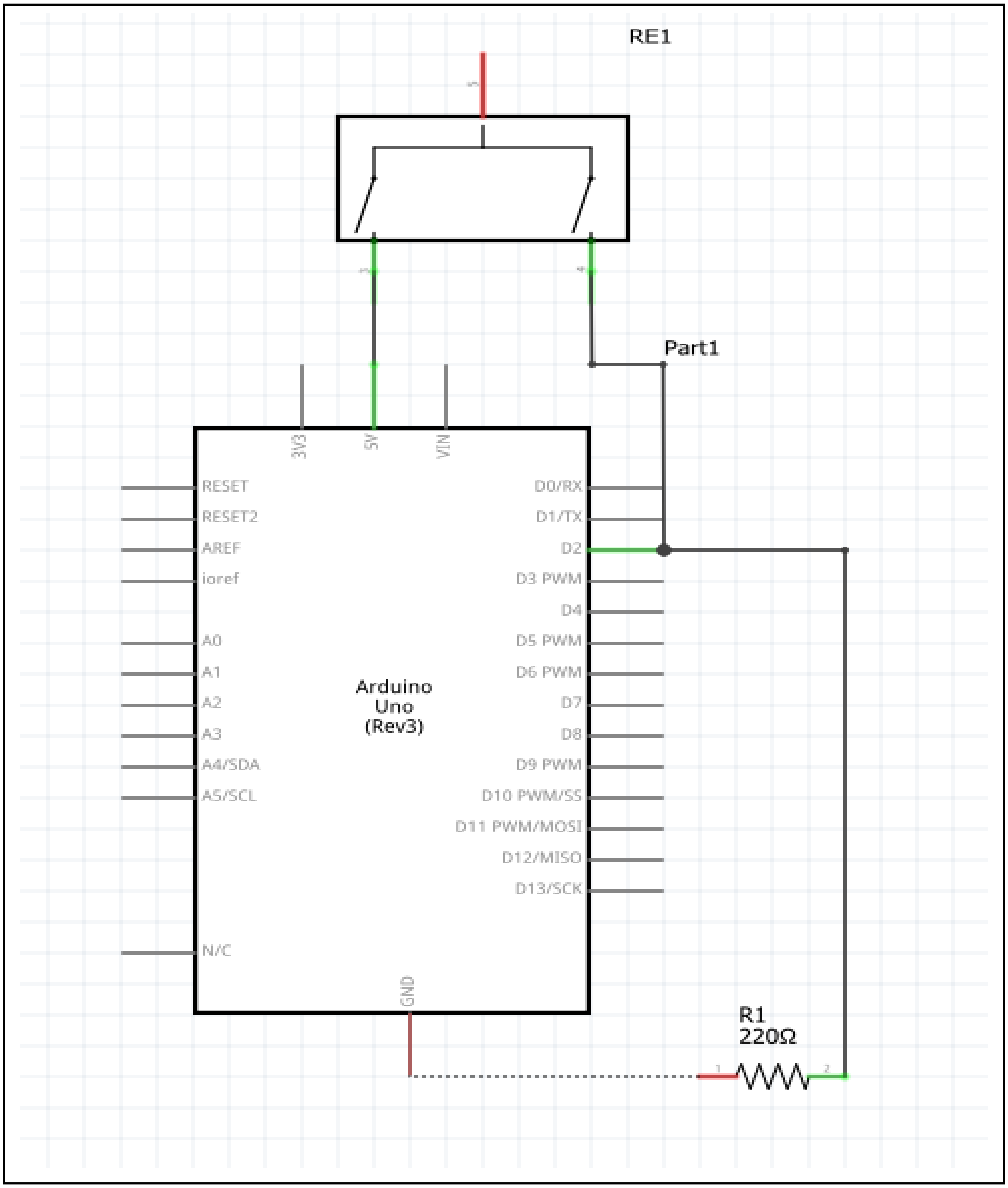
Arduino Program

```
//Name:Nayan Raj Khanal
//ID: 2227486


int binaryNumber[8];

void setup()

{

  pinMode(13, OUTPUT); // Initializing pin 13 as output pin

  pinMode(12, OUTPUT); // Initializing pin 12 as output pin

  pinMode(11, OUTPUT); // Initializing pin 11 as output pin

  pinMode(10, OUTPUT); // Initializing pin 10 as output pin

  pinMode(9, OUTPUT); // Initializing pin 9 as output pin

  pinMode(8, OUTPUT); // Initializing pin 8 as output pin

  pinMode(7, OUTPUT); // Initializing pin 7 as output pin

  pinMode(6, OUTPUT); // Initializing pin 6 as output pin

  Serial.begin(9600);

}

void displayOutput(int value) {

  for(int i = 0; i<9; i++){

      binaryNumber[i] = value % 2;

    value = value/2;

  }

  digitalWrite(13, binaryNumber[7]);

  digitalWrite(12, binaryNumber[6]);

  digitalWrite(11, binaryNumber[5]);

  digitalWrite(10, binaryNumber[4]);

  digitalWrite(9, binaryNumber[3]);

  digitalWrite(8, binaryNumber[2]);
digitalWrite(7, binaryNumber[1]);
  digitalWrite(6, binaryNumber[0]);


}


void loop()
{
  displayOutput(111);


exit(0);
}
```

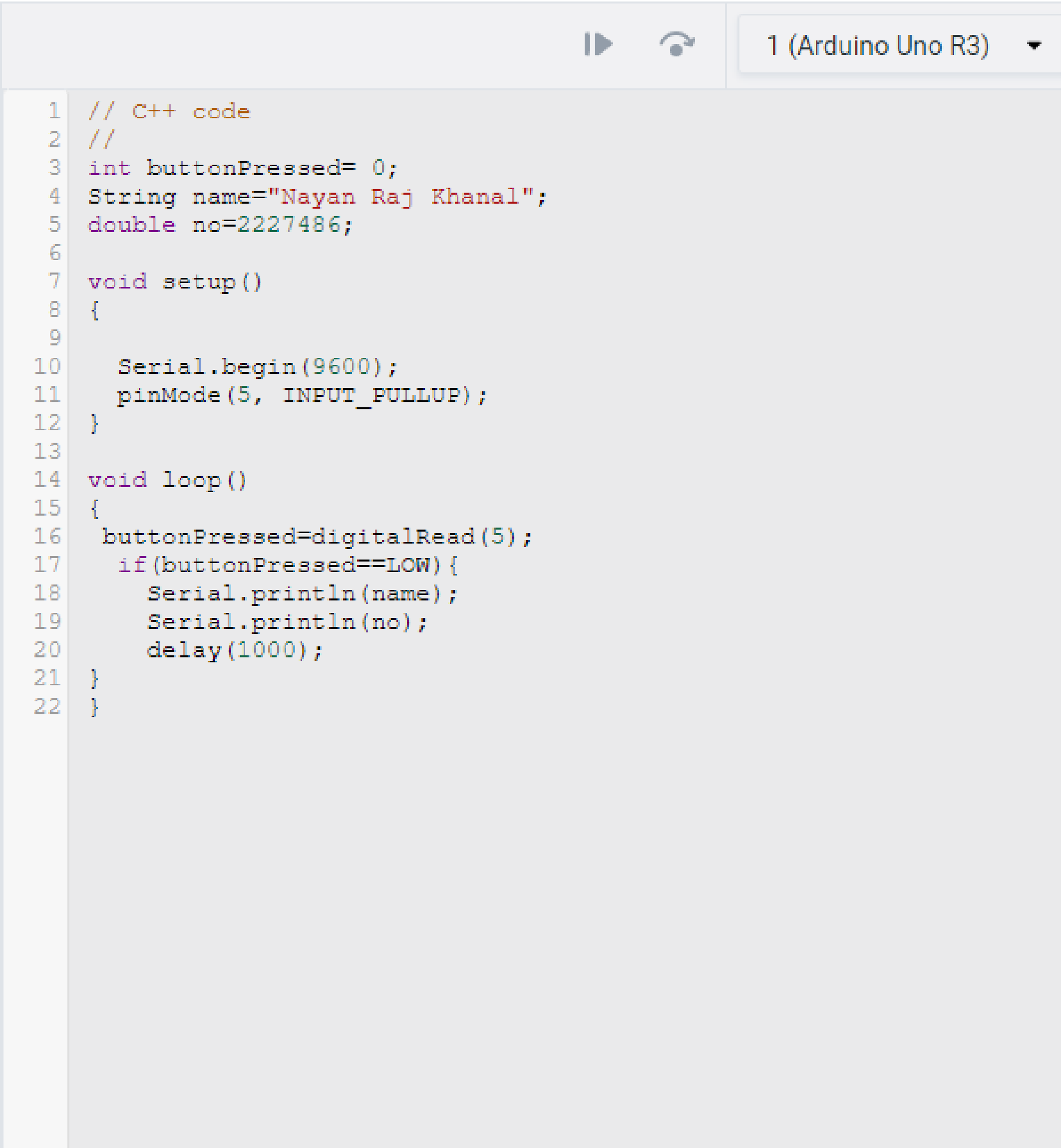
Workbook 4

Activity 4.1: Serial Port

Fritzing

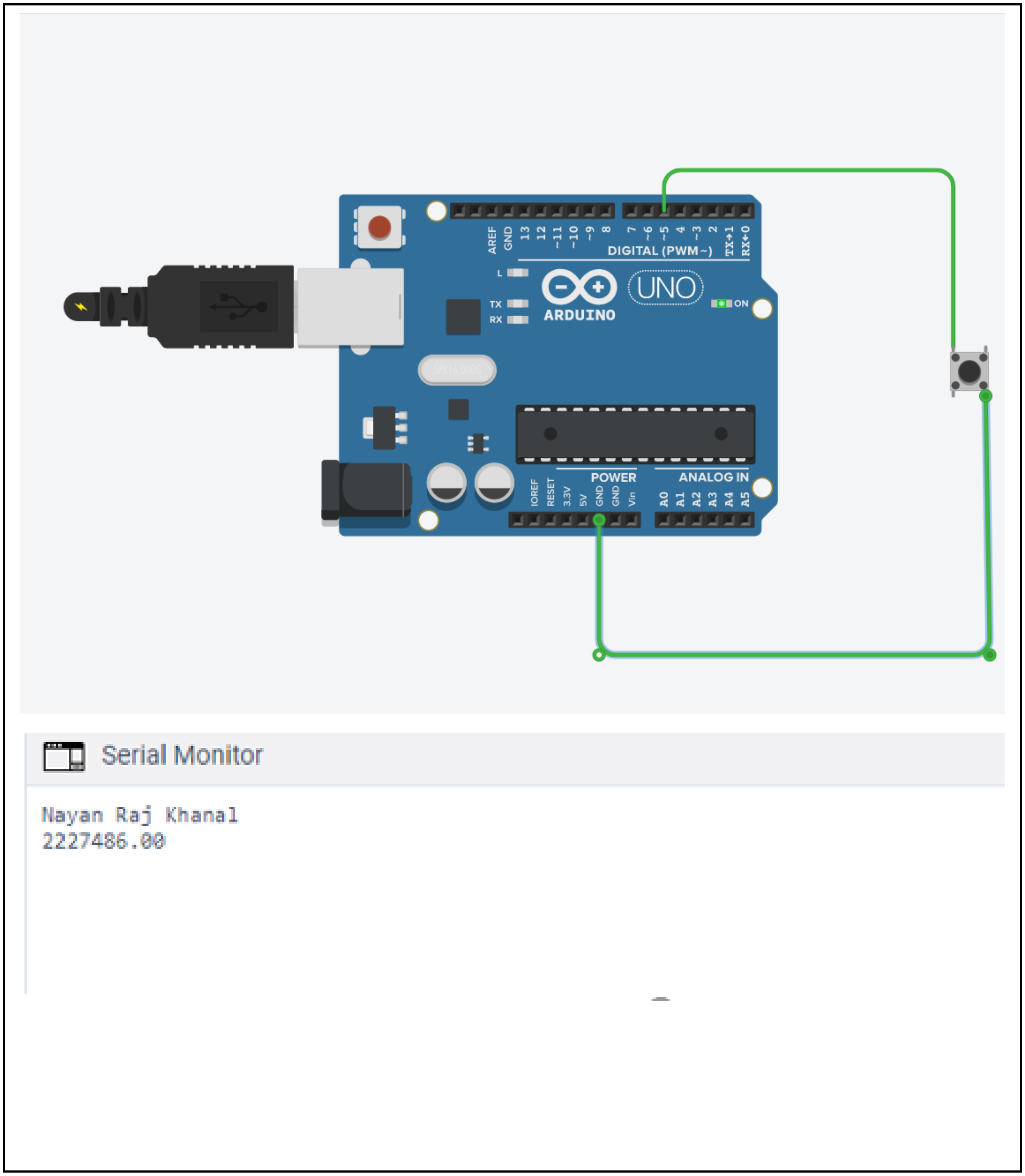


Arduino Program



```
// C++ code
//
int buttonPressed= 0;
String name="Nayan Raj Khanal";
double no=2227486;

void setup()
{

  Serial.begin(9600);
  pinMode(5, INPUT_PULLUP);
}

void loop()
{
 buttonPressed=digitalRead(5);
  if(buttonPressed==LOW){
    Serial.println(name);
    Serial.println(no);
    delay(1000);
}
}
```

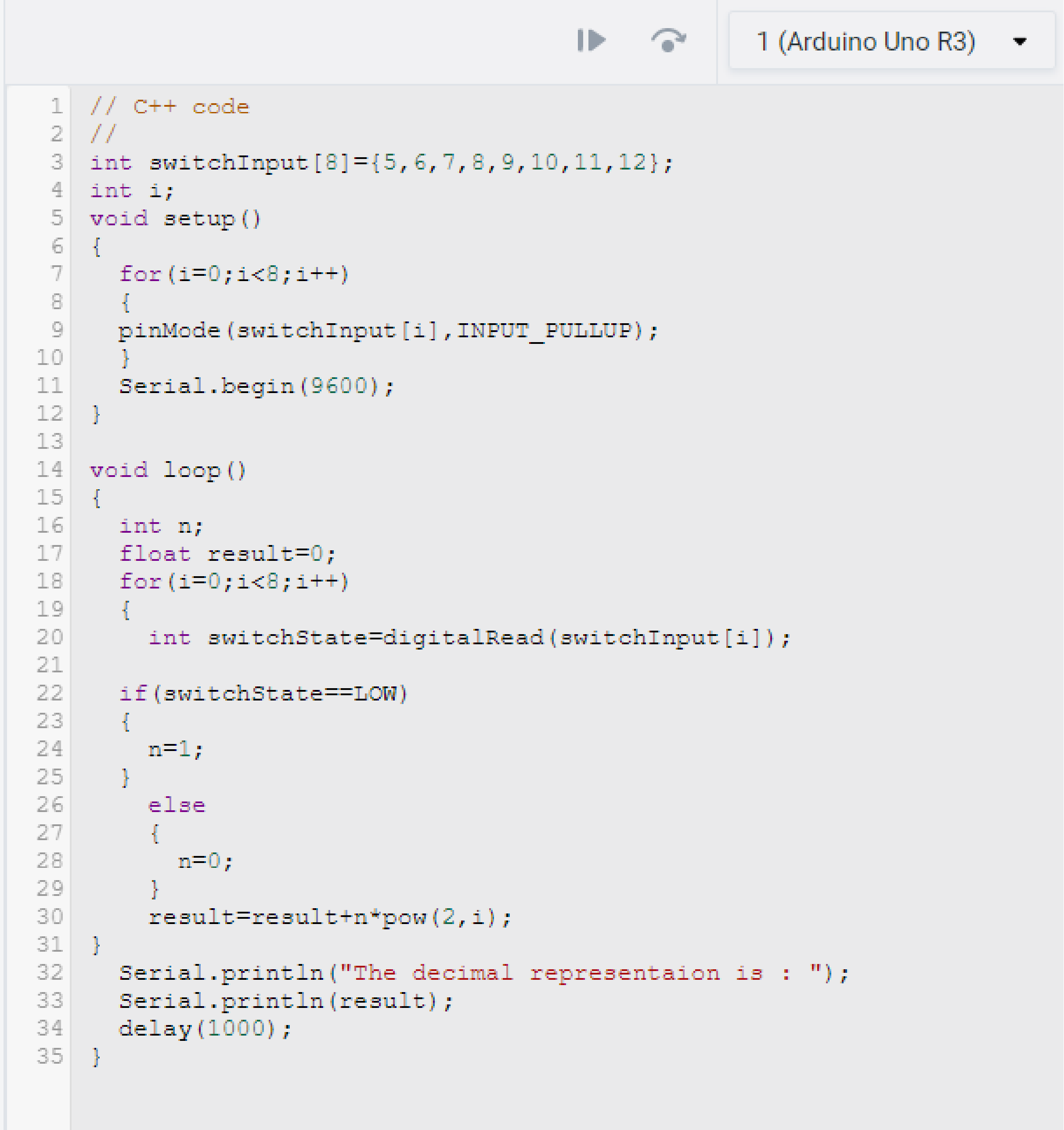
Screen Shot of Serial Port



```
Nayan Raj Khanal
2227486.00
```

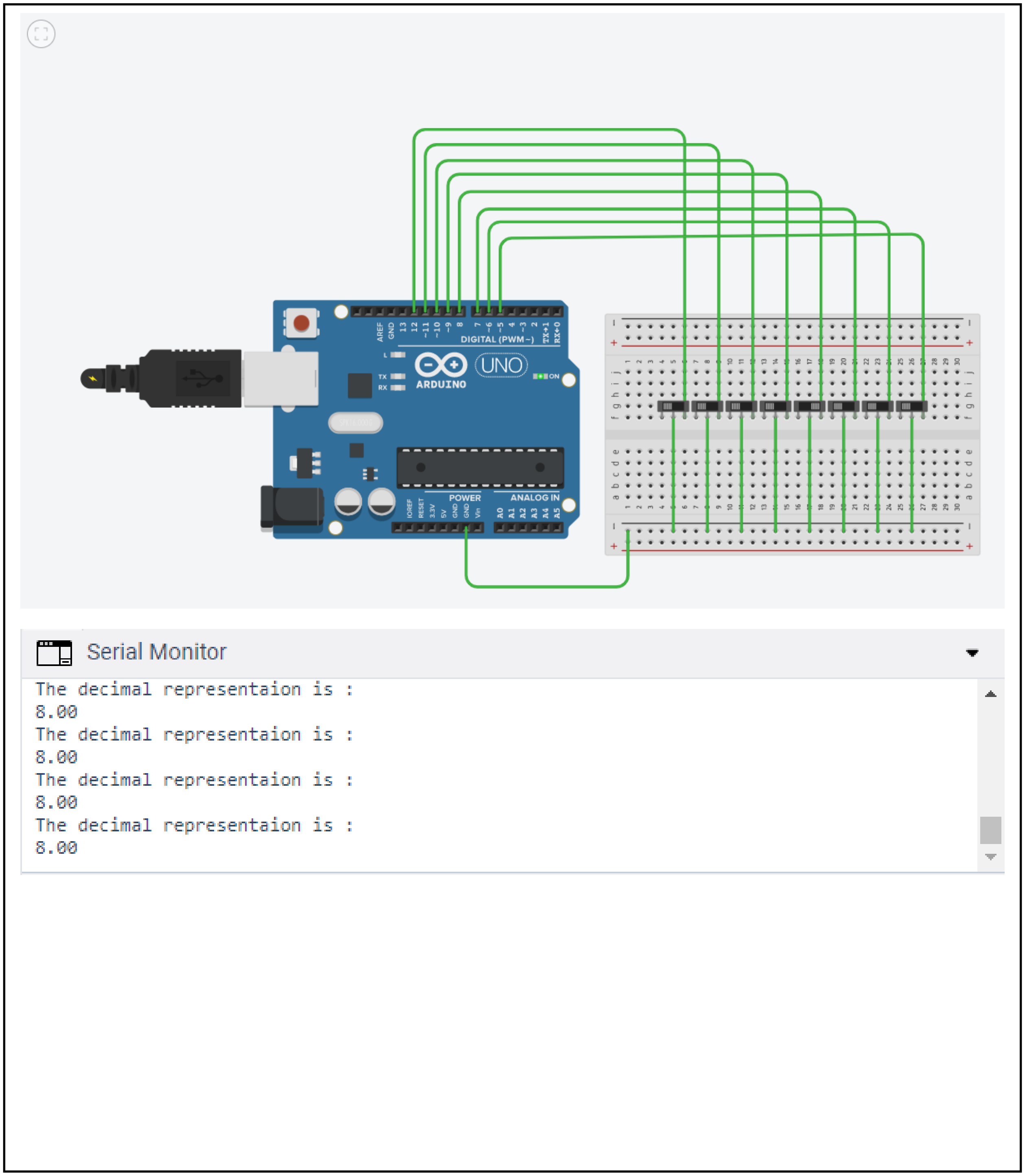
Activity 4.2: Serial Port binary to decimal

Code



```
// C++ code
//
int switchInput[8]={5,6,7,8,9,10,11,12};
int i;
void setup()
{
  for(i=0;i<8;i++)
  {
  pinMode(switchInput[i],INPUT_PULLUP);
  }
  Serial.begin(9600);
}

void loop()
{
  int n;
  float result=0;
  for(i=0;i<8;i++)
  {
    int switchState=digitalRead(switchInput[i]);

  if(switchState==LOW)
  {
    n=1;
  }
    else
    {
      n=0;
    }
    result=result+n*pow(2,i);
}
  Serial.println("The decimal representaion is : ");
  Serial.println(result);
  delay(1000);
}
```

Screen Shot of Serial Port



```
The decimal representaion is :
8.00
The decimal representaion is :
8.00
The decimal representaion is :
8.00
The decimal representaion is :
8.00
```

Activity 4.3: Calibrating Analogue Information

Code

```
//Name:Nayan Raj Khanal
//ID: 2227486


const int buttonPin=2;

int buttonState=0;


void setup()

{

 pinMode(buttonPin, INPUT); // Initializing buttonPin as input

 Serial.begin(9600);

 pinMode(A0,INPUT); // Initializing A0 as input pin

}


void loop()

{

 buttonState=digitalRead(buttonPin);

 if (buttonState==HIGH){

  int sensorValue=analogRead(A0);

  float voltage=sensorValue*(5.0/1024.0);

  Serial.print(voltage);

  Serial.println('V');

  float resistance=(voltage*250)/5;

  Serial.print(resistance);

  Serial.println("KOhm");

  delay(1000);

 }
```

Pot Resistance Clockwise

250 KOhm

Pot Resistance Anti-clockwise

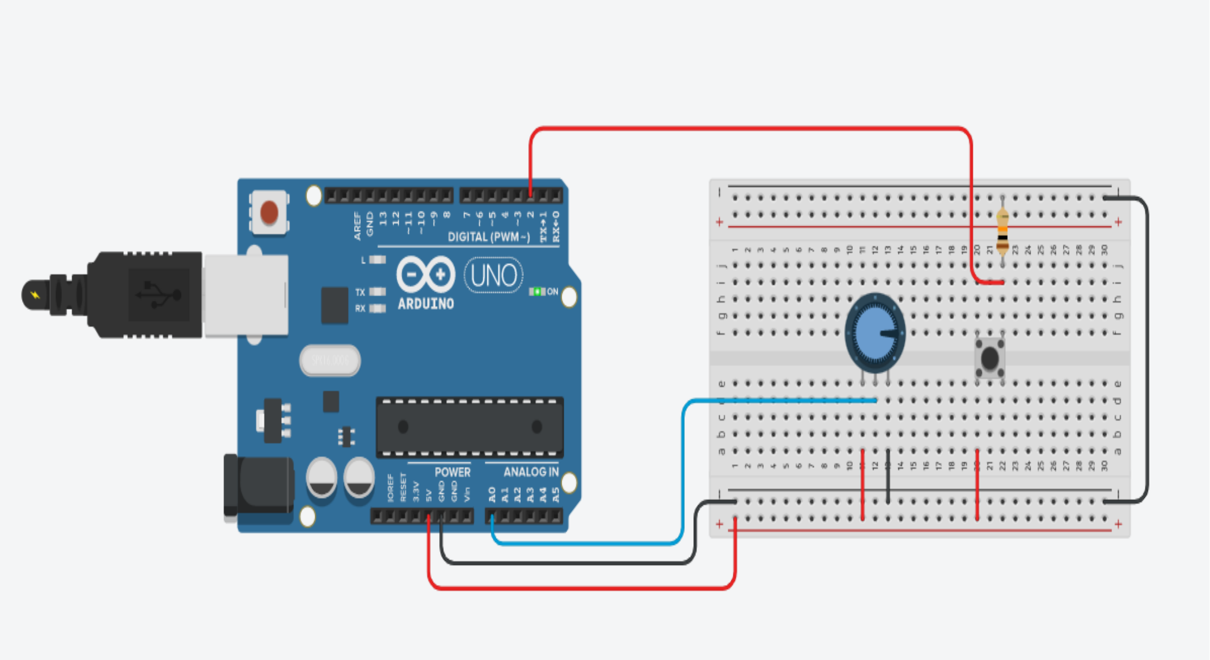
0

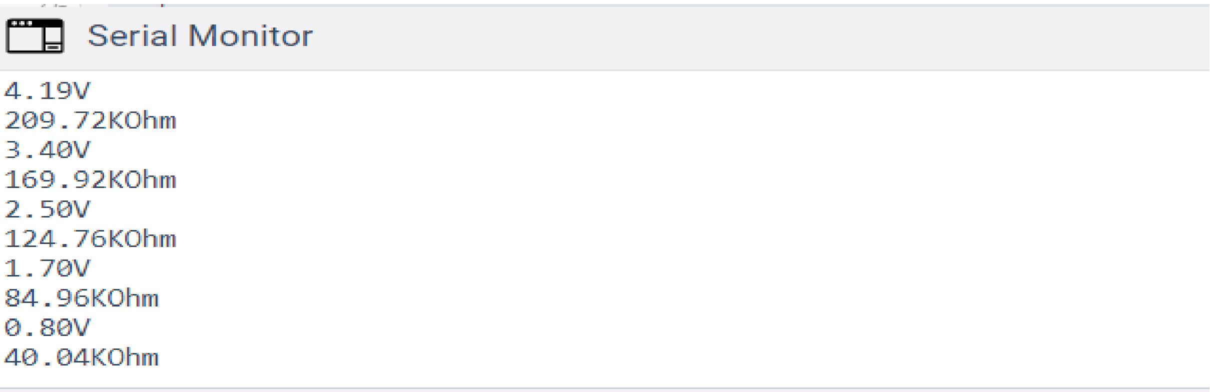
Sample of Values

Pot Resistance against Voltage change

Pot Resistance	Voltage Measured
209.72KOhm	4.19V
169.92KOhm	3.40V
124.76KOhm	2.50V
84.96KOhm	1.70V
40.04KOhm	0.80V

Screen Shot of Meaningful Serial Port Output, not just numbers





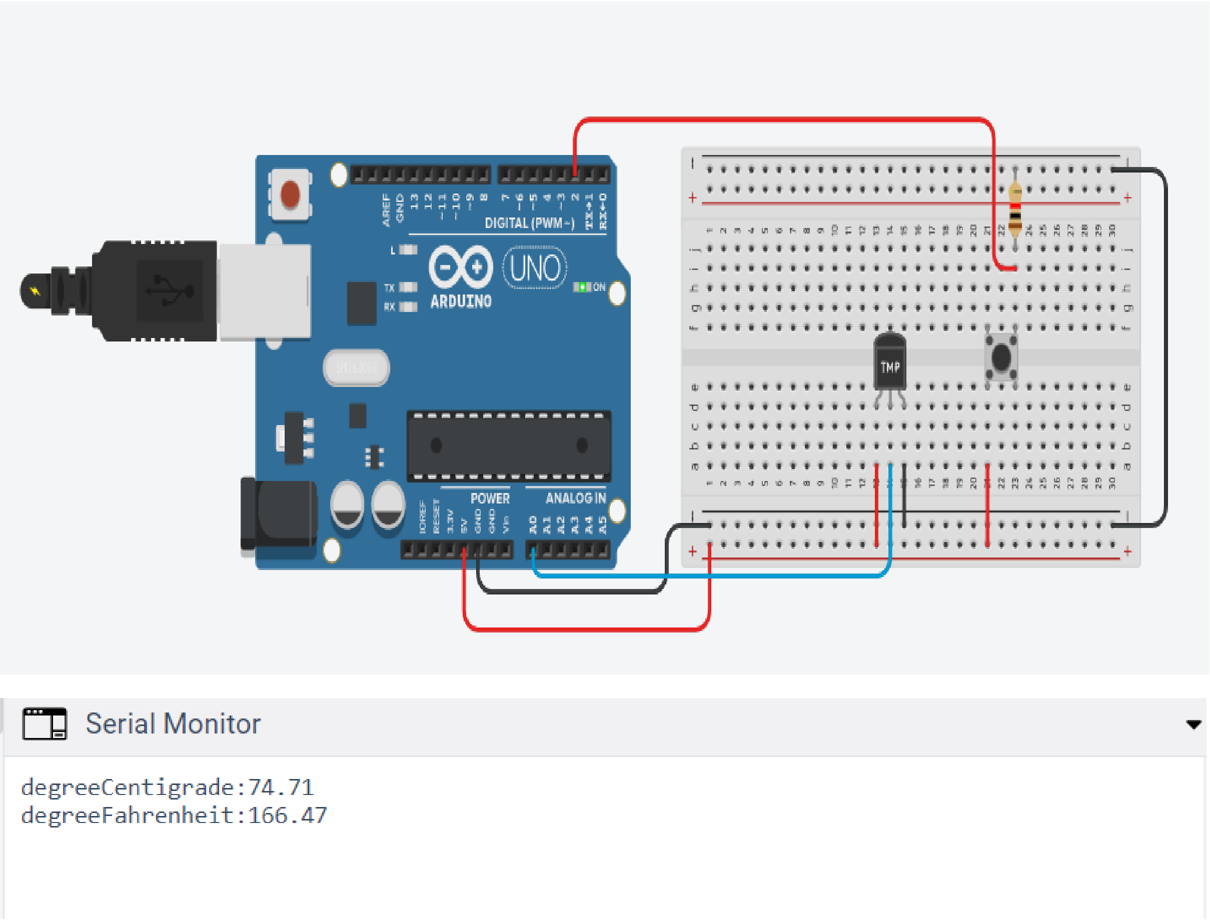
```
4.19V
209.72KOhm
3.40V
169.92KOhm
2.50V
124.76KOhm
1.70V
84.96KOhm
0.80V
40.04KOhm
```

Activity 4.4: Temperature Sensor & Serial Port

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

```
//Name:Nayan Raj Khanal
//ID: 2227486

void setup()
{
  pinMode(A0, INPUT); // Initializing A0 as input pin
  pinMode(2,INPUT); // Initializing pin 2 as input pin
  Serial.begin(9600);
}


void loop()
{
  int btn=digitalRead(2);
  if (btn==HIGH){
        float analogReading = analogRead(A0);
        float degreeCentigrade = (analogReading*500)/1024;
        Serial.print("degreeCentigrade:");
        Serial.println(degreeCentigrade);
   float fahrenheit=degreeCentigrade*1.8+32;
   Serial.print("degreeFahrenheit:");
        Serial.println(fahrenheit);
   delay(3000);
  }
}
```

Screen Shot of Serial Port

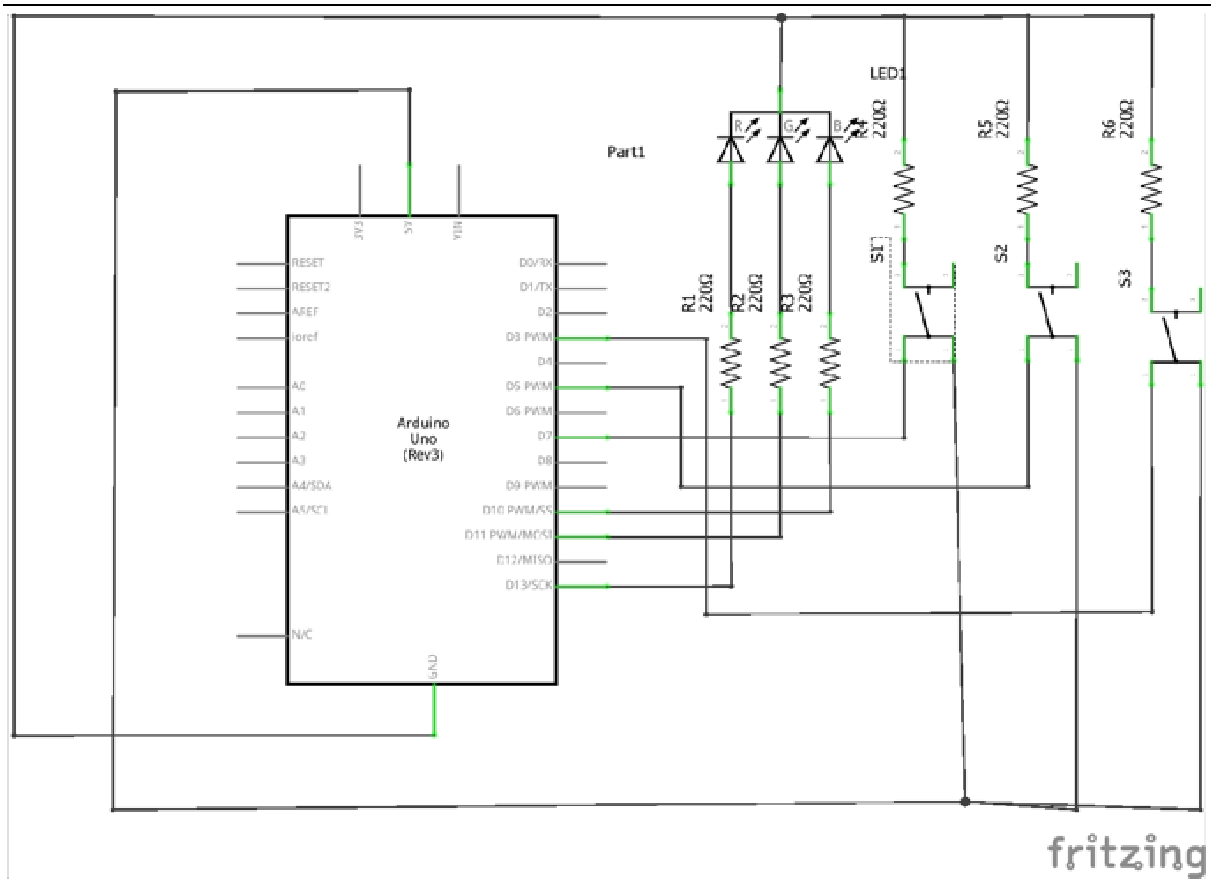


```
degreeCentigrade:74.71
degreeFahrenheit:166.47
```

Workbook 5

Activity 5.1: RGB Led and switches

Fritzing



Arduino Program

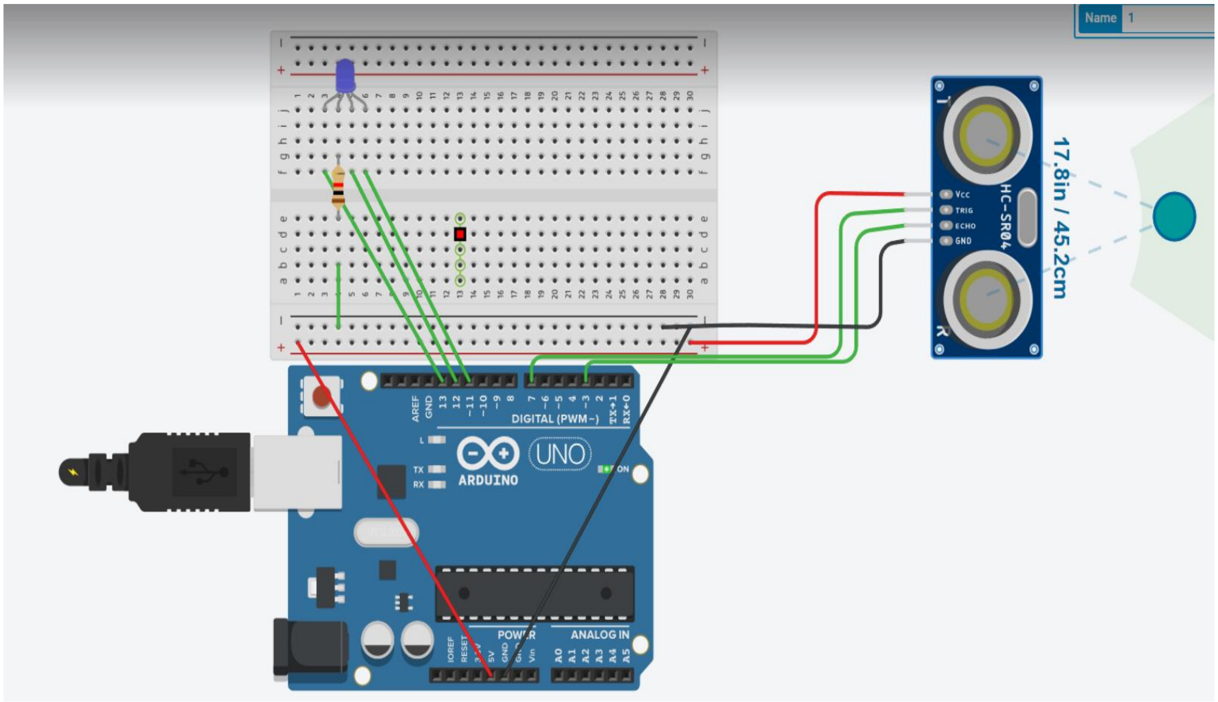
```
//Name:Nayan Raj Khanal
//ID: 2227486

// C++ code

//

// input value of three led

int red_rgb= 13;// input for red led at pin 13

int blue_rgb= 11; ;// input for blue led at pin 11

int green_rgb= 10; ;// input for green led at pin 10

 //three switch are conencted.int switchInput1=7;
  int
  switchInput2=5
  ;int
  switchInput3=3
  ;


  void setup()// starting the code

  {

  // setting pin connection to all the led

    // voltage  set for red  blue and green led

    pinMode(red_rgb,
    OUTPUT);
    pinMode(blue_rgb,
    OUTPUT);
    pinMode(green_rgb,
    OUTPUT);



    // voltage set for all the
    three switch

    pinMode(switchInput1,
    INPUT);
    pinMode(switchInput2,
    INPUT);
    pinMode(switchInput3,
    INPUT);
  }

  void loop()// starting the loop
void loop()// starting the loop
  {
int switchState1 = digitalRead(switchInput1);// input value is equal to voltage given
int switchState2 = digitalRead(switchInput2) );// input value is equal to voltage given
if (switchState1==HIGH)

{



digitalWrite(red_rgb, HIGH); // voltage setup for red led at High
delay(1000); );// wait for 1000 millisecond
digitalWrite(red_rgb, LOW); // voltage setup for redled at low
delay(1000); );// wait for 1000 millisecond }
if(switchState2==HIGH)
{
digitalWrite(blue_rgb, HIGH); // voltage setup for blue led at High
delay(1000);
digitalWrite(blue_rgb, LOW); // voltage setup for blue led at low
delay(1000); );// wait for 1000 millisecond }
if(switchState3==HIGH)
{
digitalWrite(green_rgb, HIGH); // voltage setup for green led at High
delay(1000); );// wait for 1000 millisecond
digitalWrite(green_rgb, LOW); // voltage setup for green led at low
delay(1000);// wait for 1000 millisecond
}
}
```

Activity 5.2: Distance Sensor

```
//Name:Nayan Raj Khanal
//ID: 2227486

#define trigPin 7

#define echoPin 3

int redLed=13;

int blueLed=12;

int greenLed=11;


void setup() {

    Serial.begin (9600); //view value received

    pinMode(echoPin, INPUT_PULLUP);// setting pin connection for echo pin

  pinMode(redLed, OUTPUT) );// setting pin connection for red led

  pinMode(blueLed, OUTPUT); );// setting pin connection for blue led

   pinMode(greenLed, OUTPUT); );// setting pin connection for green led


}

   void loop() {// starting the loop

    long duration, inches, cm;

    pinMode(trigPin, OUTPUT);    // calculate distance from ultrasonic pulse

    digitalWrite(trigPin, LOW);

    delayMicroseconds(2);// wait 2 microsecond

    digitalWrite(trigPin, HIGH);// voltage set for tripin

    delayMicroseconds(10);// wait for 10 macrosecond

    digitalWrite(trigPin, LOW);

    duration = pulseIn(echoPin, HIGH);

  Serial.println(duration);// serial print

    inches = duration/74/2;

  cm = duration/29/2;
```

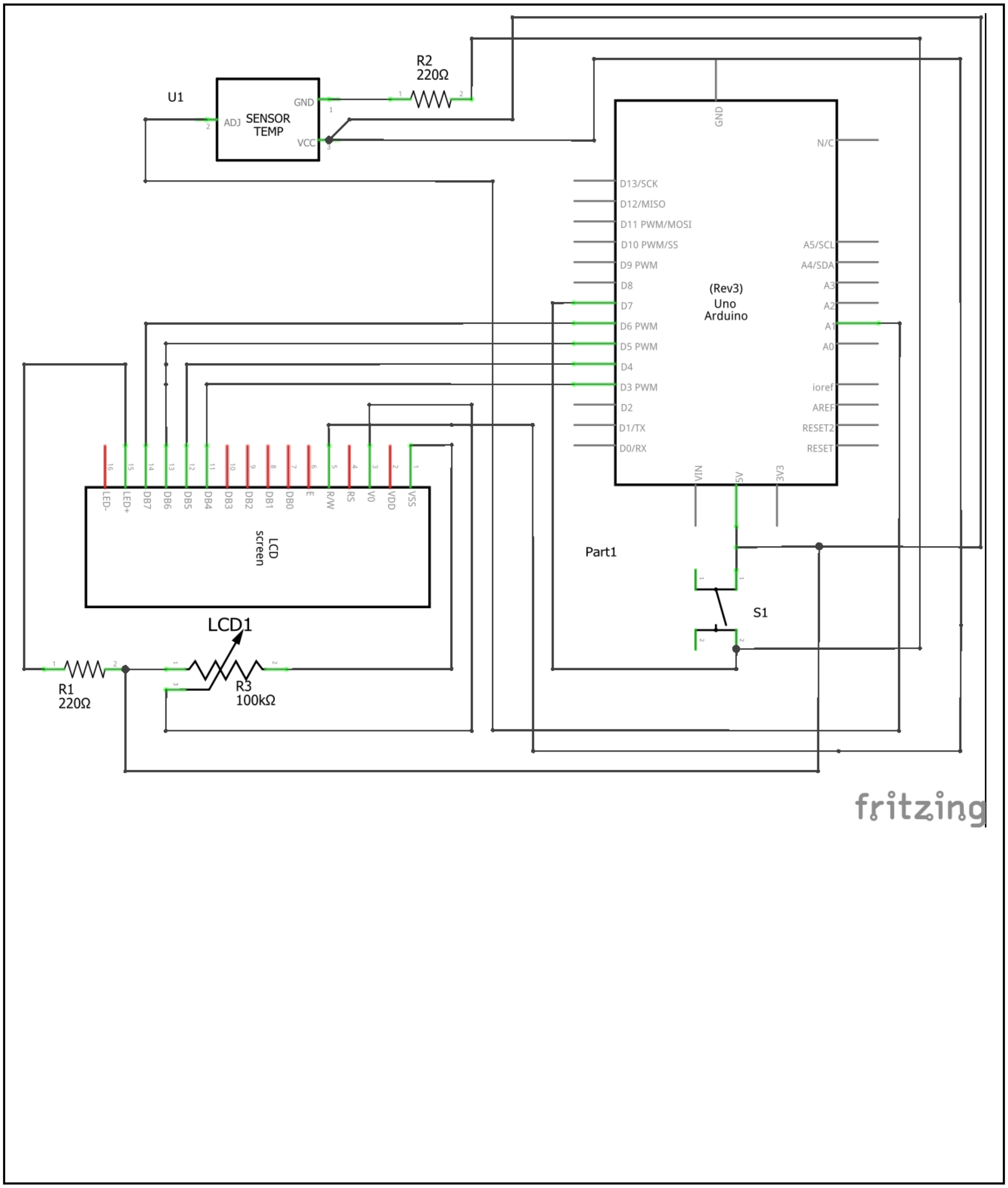
```
if(cm<200){
  if(cm<40)
  {
    digitalWrite(redLed, HIGH);// voltage set for red led
    delay(1000);// wait 1000 millisecond
    digitalWrite(redLed, LOW);
  }
  if(cm>40 && cm<100)
  {
    digitalWrite(blueLed, HIGH); );// voltage set for blue led
    delay(1000); );// wait 1000 millisecond
      digitalWrite(blueLed, LOW);
  }
   if(cm>100)
  {
    digitalWrite(greenLed, HIGH); );// voltage set for greenled
     delay(1000); );// wait 1000 millisecond
    digitalWrite(greenLed, LOW);
  }
 }
 Serial.println(cm);//view value used from ultrasonic pulse
   delay(1000);// wait for a second  1000 millisecond
}
```

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



Activity 5.3: 1602 LCD Display

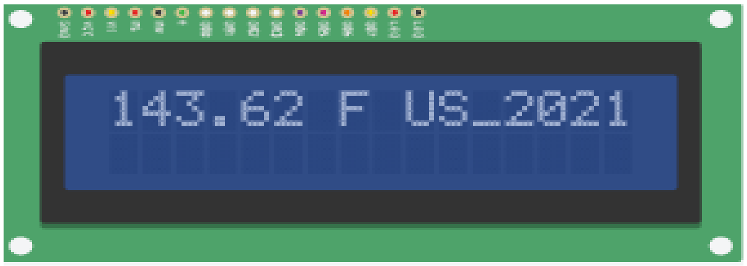
Fritzing



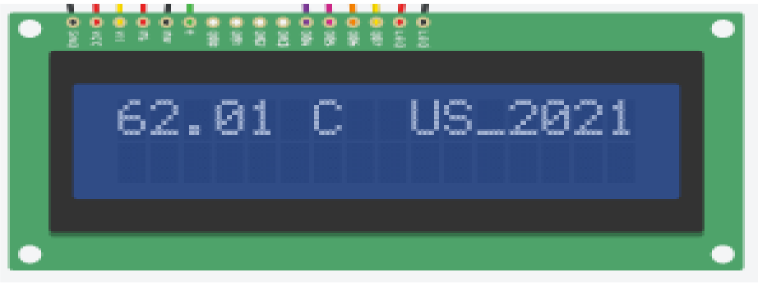
Arduino Program

```
//Name:Nayan Raj Khanal
//ID: 2227486

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int switchInput=13;

void setup() {
// Set the number of columns and rows on the LCD:
lcd.begin(16, 2);
Serial.begin(9600);
}

//Loop Method()
void loop()
{
int analogReading = analogRead(A0);
float degreeC = analogReading*500.0/1024.0;
float degreeF = (degreeC*9.0/5.0)+32;
int switchState = digitalRead(switchInput);
Serial.println(switchState);

//Using if and else condition:
if (switchState == LOW)
{
lcd.print(degreeC);
lcd.print(" C");
lcd.setCursor(9,0);
lcd.print("US_2021");
delay(1000);
lcd.clear();
}
else
{
lcd.print(degreeF);
lcd.print(" F");
lcd.setCursor(9,0);
lcd.print("US_2021");
delay(1000);
lcd.clear();
}
```

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



```
143.62 F US_2021
```

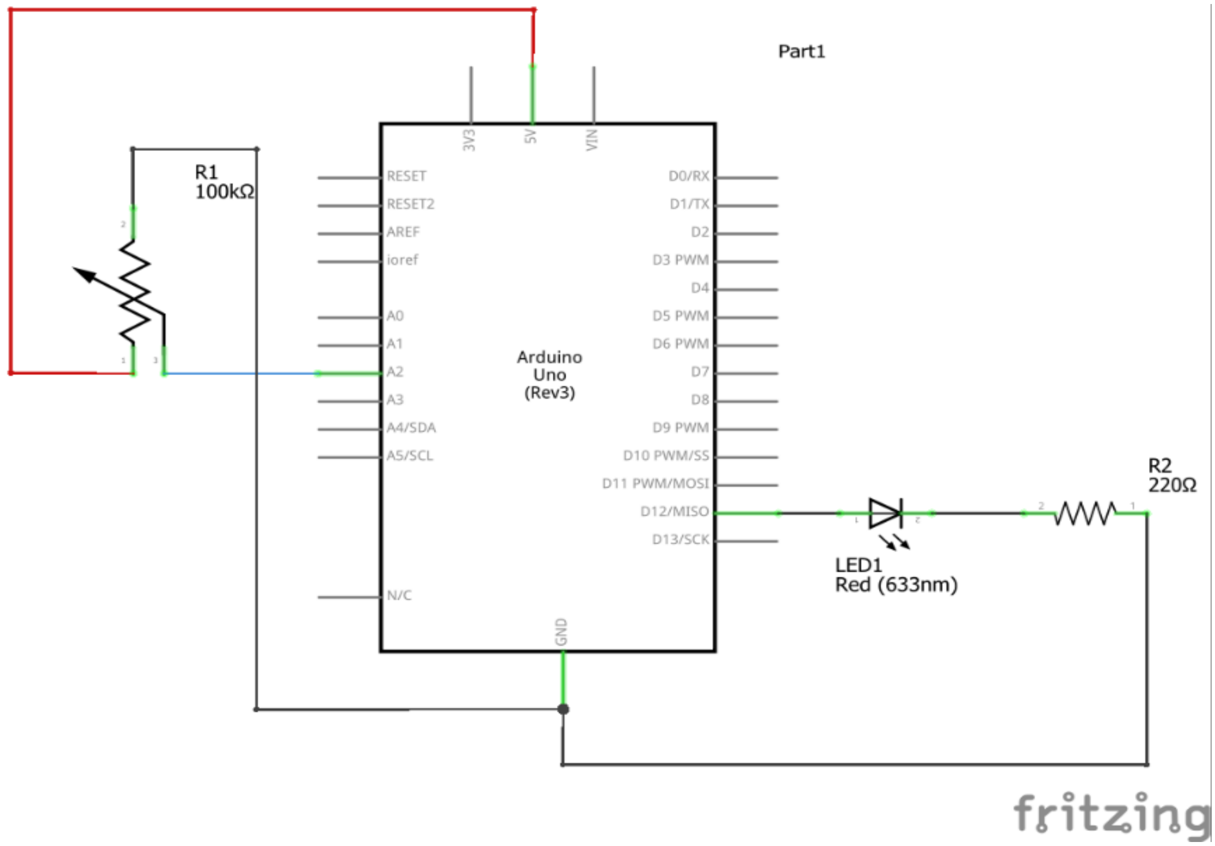


```
62.01 C  US_2021
```

Workbook 6

Activity 6.1: PWM

Fritzing



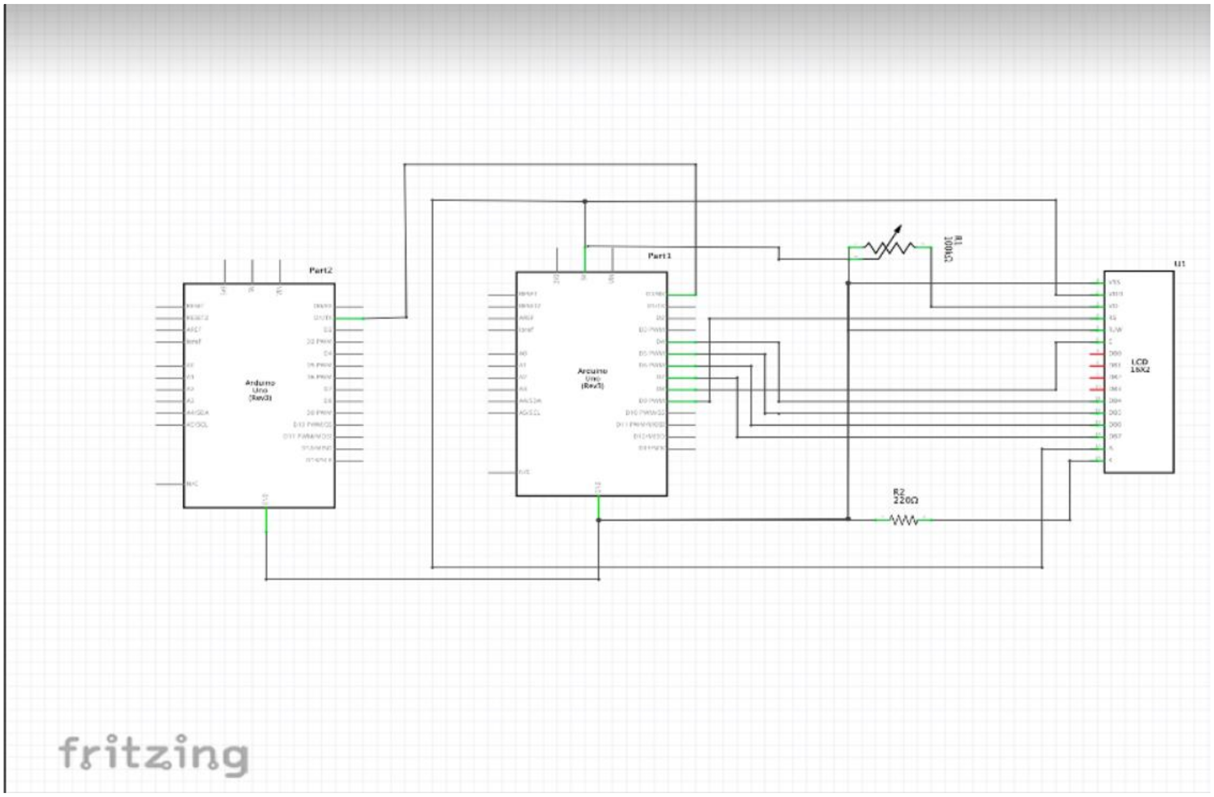
```
//Name:Nayan Raj Khanal
//ID: 2227486

int Led = 12; //Declaring variables
int pot = A2;
void setup() //Declaring variables and assigned by setup method()
{
pinMode(pot, INPUT);
pinMode(Led, OUTPUT);
Serial.begin(9600);
}
void loop() //Loop method
{
int voltage = (255./1023.) * potRead(A1);
Serial.println(voltage);
potWrite(Led, voltage);
}
```

Arduino Program

Workbook 7

Activity 7.1: 2 Arduinos – using Digital Pins

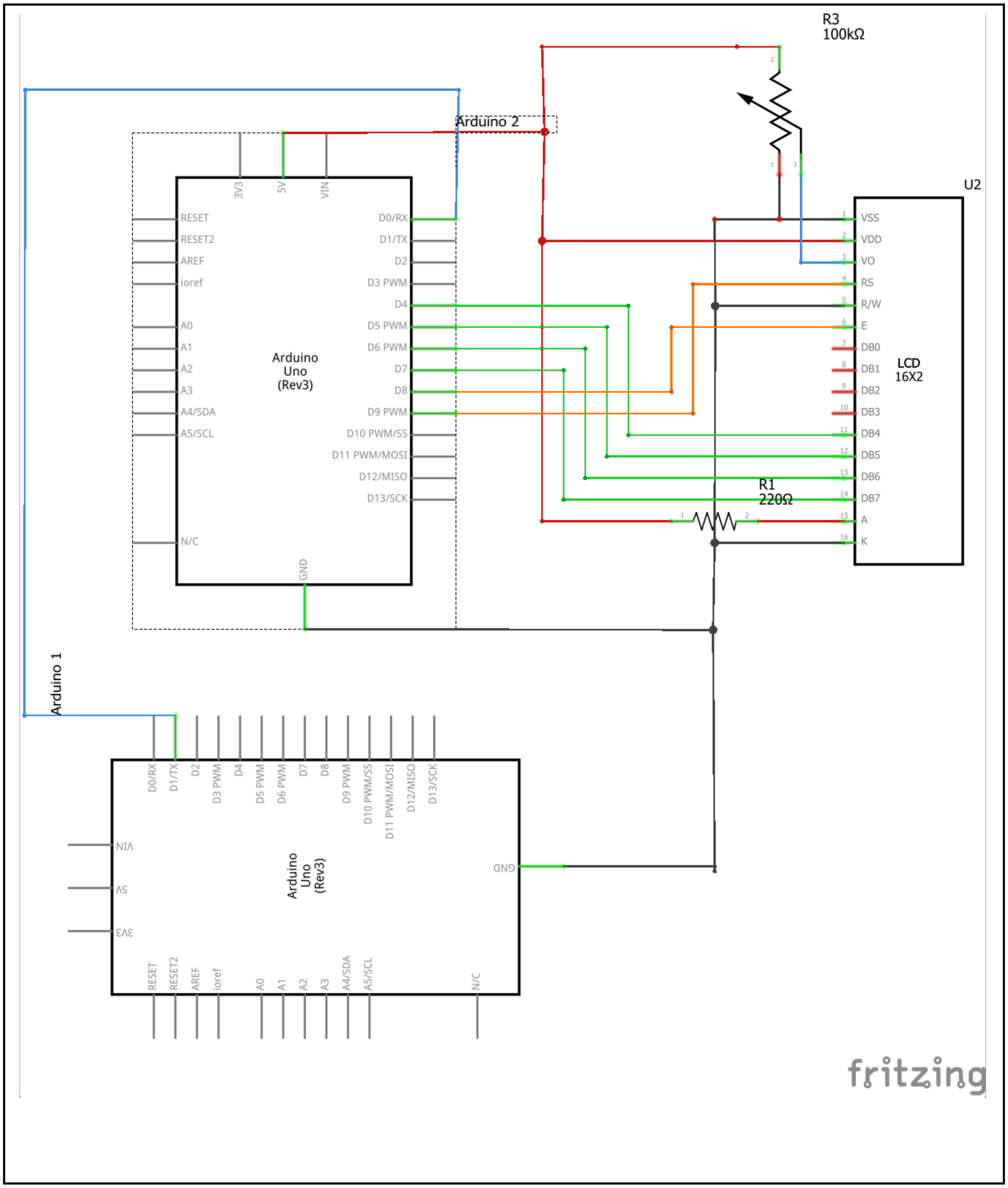
Fritzing



```
//Name:Nayan Raj Khanal
//ID: 2227486

// include the library code:
#include <LiquidCrystal.h>
char test[10];// input the value
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);//// initialize the library with the numbers of the interface
pins
void setup() {// staring the program
lcd.begin(16, 2);// lcd communication begin
Serial.begin(9600);// serial communciation begin
}
void loop() {
Serial.readBytes(test,6);
lcd.print(test);// print the value i lcd
delay(50);// wait for 50 millisecond
lcd.clear();// clear the lcd
}


Arduino 2:
char a[10];// input of value
void setup() {// staring the program
Serial.begin(9600);// serial communciation begin
}
void loop() {//starting the loop
while(Serial.available()>0){
int i = 0;
char c = Serial.read();// value input is equal to value in cd
if(c!='\n'){
a[i]=c;
i++;
}
Serial.write(a, 1);
}
}
```

Activity 7.2: 2 Arduinos – using Serial I/O

Fritzing

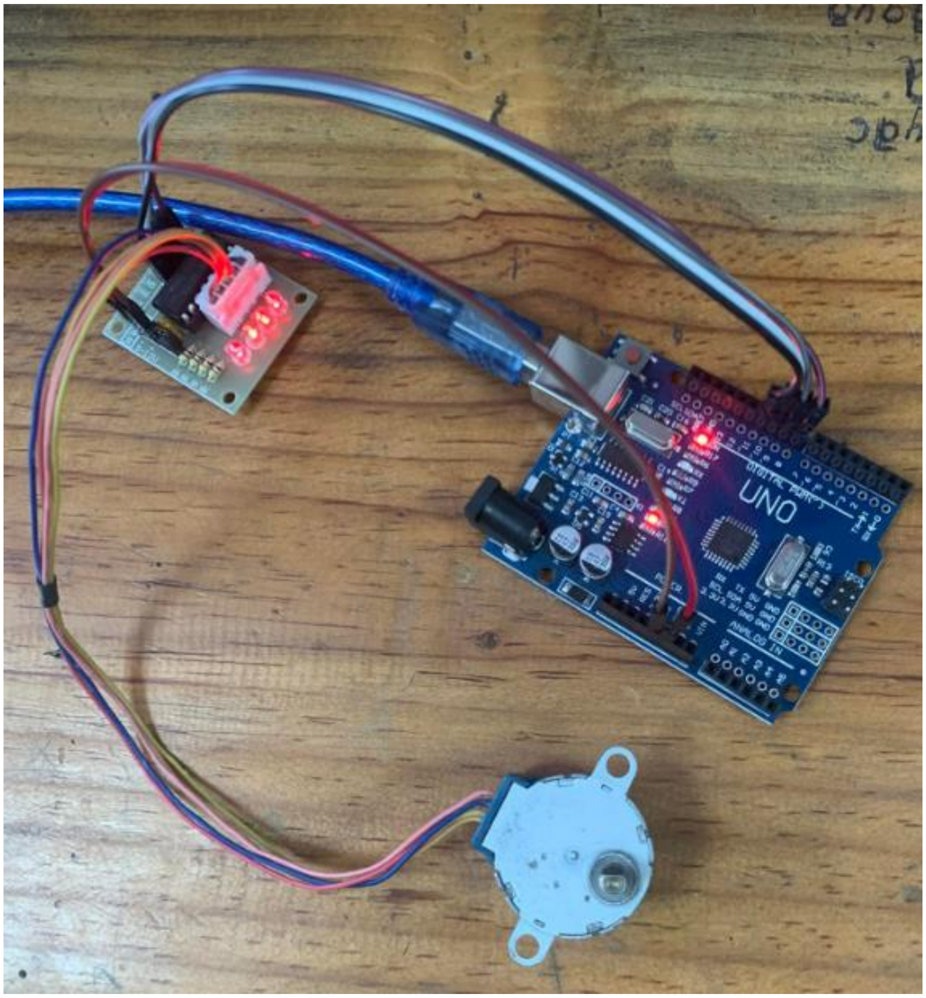


Arduino Program

```
//Name:Nayan Raj Khanal
//ID: 2227486


// include the library code:
#include <LiquidCrystal.h>
char test[10];// input the value
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);//// initialize the library with the numbers of the interface pins
void setup() {// staring the program
lcd.begin(16, 2);// lcd communication begin
Serial.begin(9600);// serial communciation begin
}
void loop() {
Serial.readBytes(test,6);
lcd.print(test);// print the value i lcd
delay(50);// wait for 50 millisecond
lcd.clear();// clear the lcd
}
Arduino 2:
// student name: Kaushal Rai
// Student ID: 2226684
char name[10]="Kaushal" ;
void setup() {
Serial.begin(9600);
}
void loop() {
Serial.write(name,6);
delay(100);
}
```

Workbook 8

Activity 8.1: Stepper Motor Circuit Diagram



Circuit Diagram

Arduino Program

```
//Name:Nayan Raj Khanal
//ID: 2227486

#include <AccelStepper.h>
#define HALFSTEP 8
// Motor pin definitions
#define motorPin1  8     // IN1 on the ULN2003 driver 1
#define motorPin2  9     // IN2 on the ULN2003 driver 1
#define motorPin3  10     // IN3 on the ULN2003 driver 1
#define motorPin4  11     // IN4 on the ULN2003 driver 1

// Initialize with pin sequence IN1-IN3-IN2-IN4 for using the AccelStepper with 28BYJ-48
AccelStepper stepper1(HALFSTEP, motorPin1, motorPin3, motorPin2, motorPin4);

void setup() {
  stepper1.setMaxSpeed(2000.0);
  stepper1.setAcceleration(1000.0);
  stepper1.setSpeed(2000);
  stepper1.moveTo(2000);
}//--(end setup )---

void loop() {

  //Change direction when the stepper reaches the target position
  if (stepper1.distanceToGo() == 0) {
    stepper1.moveTo(-stepper1.currentPosition());
  }
  stepper1.run();
}
```

Activity 8.2: 2 Stepper Motors

Arduino Program

```
//Name:Nayan Raj Khanal
//ID: 2227486

#include <Stepper.h>

#include <AccelStepper.h>

#define HALFSTEP 8

int Pin1 = 8;//IN1 is connected to 8

int Pin2 = 9;//IN2 is connected to 9

int Pin3 = 10;//IN3 is connected to 10

int Pin4 = 11;//IN4 is connected to 11

int switchCW  =2;

int switchCCW =3;

int pole1[] ={0,0,0,0, 0,1,1,1, 0};

int pole2[] ={0,0,0,1, 1,1,0,0, 0};

int pole3[] ={0,1,1,1, 0,0,0,0, 0};

int pole4[] ={1,1,0,0, 0,0,0,1, 0};

int poleStep = 0;

int  dirStatus = 3;


void setup()

{

 pinMode(Pin1, OUTPUT);//define pin for  in1

 pinMode(Pin2, OUTPUT);//define pin for in2

 pinMode(Pin3, OUTPUT);//define pin for  in3

 pinMode(Pin4, OUTPUT);//define pin for  in4


 pinMode(switchCW,INPUT_PULLUP);// CW (Clock Wise)push button pin as input

 pinMode(switchCCW,INPUT_PULLUP);//CCW(Anti ClockWise) push button pin as input

}
```

```
void loop(){
  if(digitalRead(switchCCW) == LOW) {
    dirStatus =1;
  }else if(digitalRead(switchCW) == LOW) {
   dirStatus  = 2;
  }
 if(dirStatus ==1){
   poleStep++;
    driveStepper(poleStep);
 }else if(dirStatus ==2){
   poleStep--;
    driveStepper(poleStep);
 }else{
  driveStepper(8);
 }
 if(poleStep>7){
   poleStep=0;
 }
 if(poleStep<0){
   poleStep=7;
 }
 delay(1);
}
void driveStepper(int c)
{
     digitalWrite(Pin1, pole1[c]);
     digitalWrite(Pin2, pole2[c]);
     digitalWrite(Pin3, pole3[c]);
     digitalWrite(Pin4, pole4[c]);
}
```

Workbook 9

Activity 9.1: Windscreen Wiper Code using Servos & Temperature Sensor

```
//Name:Nayan Raj Khanal
//ID: 2227486

#include<Servo.h>  //importing library

//creating object for two servo to control

Servo servo1;

Servo servo2;

int pos=0;    //initial position value 0 for servo


void setup ()

{

servo1.attach(6);  //servo1 connected to 6 no. digital pin

servo2.attach(9);  //servo2 connected to 9 no. digital pin

  Serial.begin(9600);  //Serial monitor enabled

}

void loop()

{

int sensor=analogRead(A0);  //variable for reading Analog data of temperature sensor

  float cel=sensor*500/1023.0;  //calculating temperature using formula

  Serial.println(cel); //printing output in serial monitor

  if (cel>=20){ //if condition satisfied, codes in block below runs

     for(pos=0;pos<180;pos++)

  { //rotates both servo at 180 degree from 0

    servo1.write(pos);

    servo2.write(pos);

    delay(30); }
for(pos=180;pos>0;pos--)
  { //rotates both servo at 0 degree from 180
    servo1.write(pos);
    servo2.write(pos);
    delay(30);
  }
  }
  else{ //condition not satisfied servo stays at initial position
    servo1.write(pos);
    servo2.write(pos);
    delay(50);
  }
}
```

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.