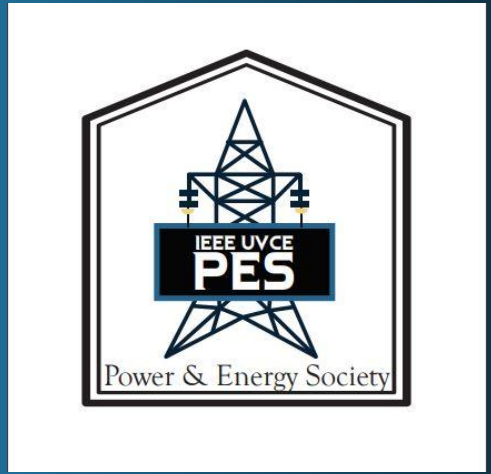
VOLTORB

JANUARY

2023



Problem Statement

Vivek is a common man in a country where there's lot of accidents taking place daily. The traffic police there aren't caring about happening. Vivek research finds that there is no traffic signal due to which many accidents are taking place. He wants your help now to make it and help many lives to survive. Can you help Vivek by building a manually operated traffic signal?

Instructions:

- The device must contains 3 different lights [red yellow, green].
- There must be a sound when device has red light.
- There can be single/multiple circuits.
- External creativity like Solar panel and Light sensors will add points to it.

COMPONENTS REQUIRED

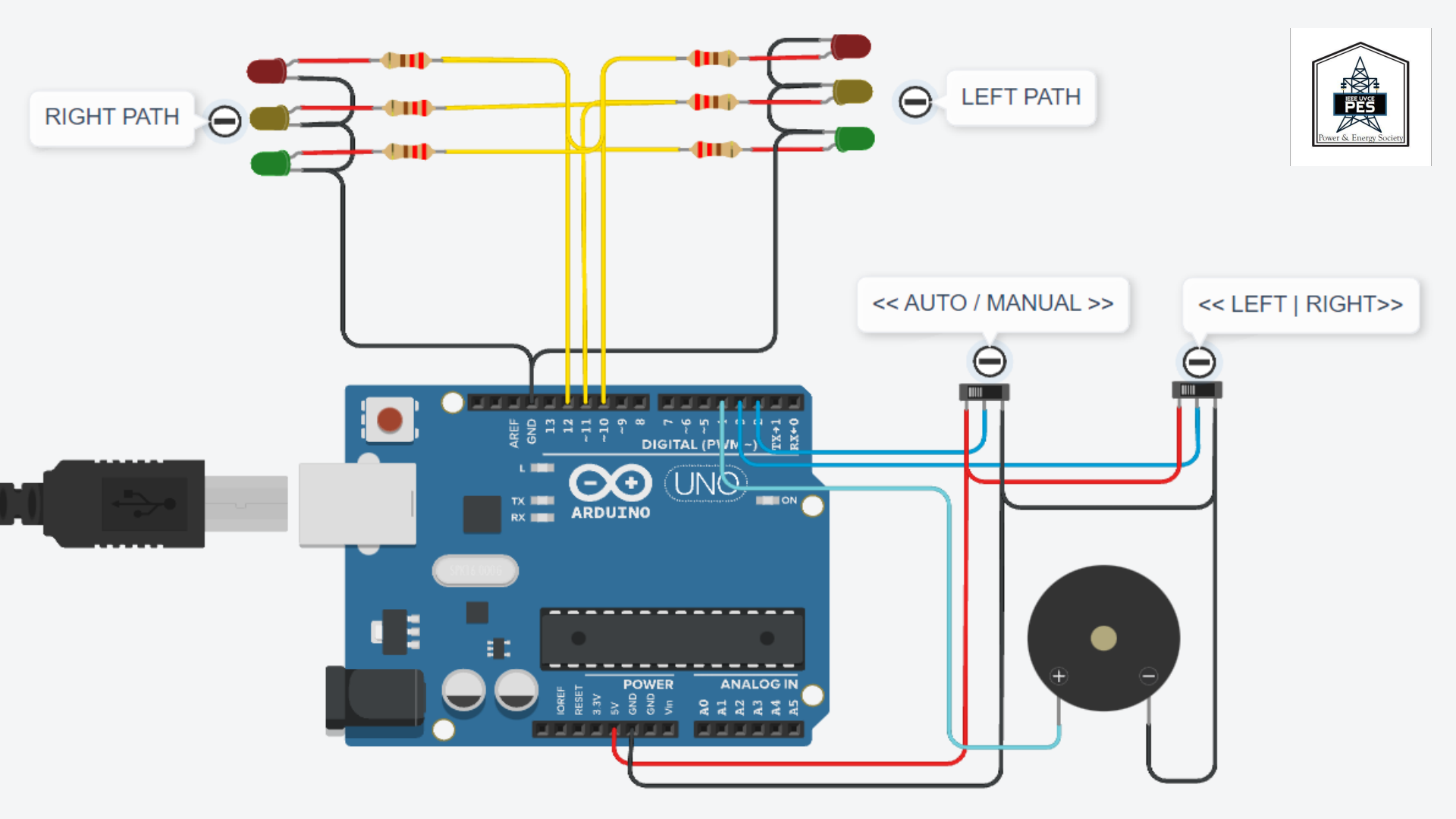
- Arduino UNO (controller)
- LEDs (RED, YELLOW, GREEN) for signal lights
- Toggle Switch SPDT (Single pole Dual thro) as user Inputs
- Buzzer

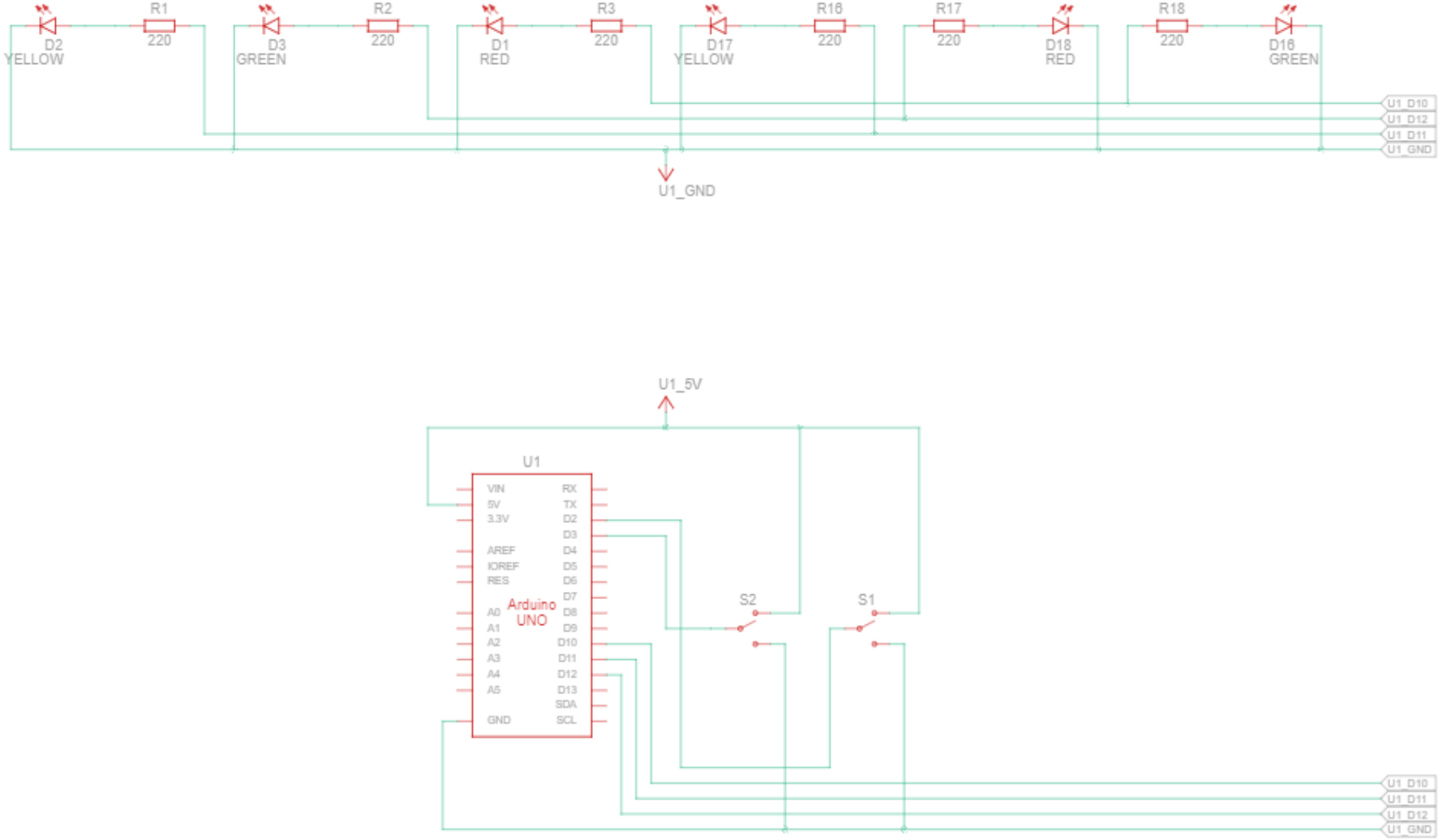
Arduino UNO

LEDs

Switch

Buzzer


RIGHT PATH
LEFT PATH
<< AUTO / MANUAL >>
<< LEFT | RIGHT>>
IEEE UVCE
PES
Power & Energy Society
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5


R1
220
D2
YELLOW
R2
220
D3
GREEN
R3
220
D1
RED
R16
220
D17
YELLOW
R17
220
D18
RED
R18
220
D16
GREEN
U1_D10
U1_D12
U1_D11
U1_GND
U1_GND
U1_5V
U1
VIN
5V
3.3V
AREF
IOREF
RES
A0
A1
A2
A3
A4
A5
GND
Arduino
UNO
RX
TX
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11
D12
D13
SDA
SCL
S2
S1
U1_D10
U1_D11
U1_D12
U1_GND

Program Algorithm

- Initialize serial, input and output
- Beep Buzzer while every transition
- While **MANUAL** mode is selected i.e. Pin 2 is LOW
  - Red at right and Green at left if **LEFT** is selected
  - Red at left and Green at right if **RIGHT** is selected

- While **AUTO** mode is selected i.e. Pin 2 is HIGH
  - Yellow for few seconds
  - Red at right and Green at left for set Time
  - Yellow for few seconds
  - Green at right and Red at left for set Tim

```
// C++ code
// IEEE UVCE PES
// VOLTORB JAN 2023
void setup(){
  Serial.begin(9600);
  pinMode(12,OUTPUT);
  pinMode(11,OUTPUT);
  pinMode(10,OUTPUT);

  pinMode(3,INPUT);
  pinMode(2,INPUT);
  Serial.println("IEEE UVCE PES");
  tone(4,1000,1000);
  delay(700);
  //noTone(4);
}

void loop(){
  while(digitalRead(2)){ // WHILE MODE IS AUTO
    Serial.println("AUTO MODE");
    //YELLOW
    digitalWrite(12,LOW);
    digitalWrite(11,HIGH);
    digitalWrite(10,LOW);
    delay(500);
    //RIGHT
    tone(4,1000,800);
    digitalWrite(12,LOW);
    digitalWrite(11,LOW);
    digitalWrite(10,HIGH);
    delay(5000);
    //YELLOW
    digitalWrite(12,LOW);
    digitalWrite(11,HIGH);
    digitalWrite(10,LOW);
    delay(500);
    //LEFT
    tone(4,1000,800);
    digitalWrite(12,HIGH);
    digitalWrite(11,LOW);
    digitalWrite(10,LOW);
    delay(5000);
  }// auto mode end
  while(!digitalRead(2)){
    Serial.println("MANUAL");
    //YELLOW
    digitalWrite(12,LOW);
    digitalWrite(11,HIGH);
    digitalWrite(10,LOW);
    delay(500);
    while(digitalRead(3) && !digitalRead(2)){ // if left
      Serial.println("LEFT MANUAL");
        //LEFT
        tone(4,1000,800);
        digitalWrite(12,HIGH);
        digitalWrite(11,LOW);
        digitalWrite(10,LOW);
    }
    //YELLOW
    digitalWrite(12,LOW);
    digitalWrite(11,HIGH);
    digitalWrite(10,LOW);
    delay(500);
    while(!digitalRead(3)&& !digitalRead(2)){ //if right
        //RIGHT
        Serial.println("RIGHT MANUAL");
        tone(4,1000,800);
        digitalWrite(12,LOW);
        digitalWrite(11,LOW);
        digitalWrite(10,HIGH);
    }
  }
}
```

Program in GitHub repository

THANK YOU

IEEE UVCE PES

TinkerCad simulation Link