VISVESVARAYA NATIONAL INSTITUTE OF TECHNOLOGY (VNIT), NAGPUR

Micro-Controller Project (EEL330)

Submitted by:
Soumyadeep Ghosh (BT20ECE036)
Kirit Manchare (BT20EEE065)
Pranjali Kulkarni (BT20EEE087)
Shubham Khune (BT20EEE095)
Vedant Goyal (BT20EEE110)
Semester 6

Submitted to:
Dr. Bhushan Rajpathak
(Course Instructor)
Department of Electrical and Electronics Engineering,VNIT Nagpur

CONTENTS

INTRODUCTION:-

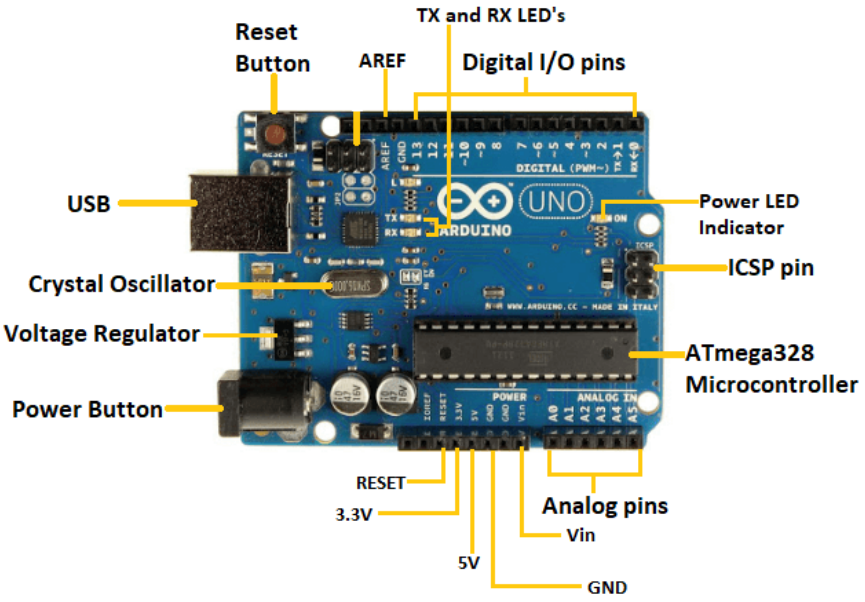
Driving while drowsy or fatigued is a major cause of road accidents around the world. According to a report by the National Highway Traffic Safety Administration (NHTSA), drowsy driving causes an estimated 100,000 crashes, 71,000 injuries, and 1,550 fatalities each year in the United States alone. To address this issue, various technologies have been developed to help drivers stay alert and prevent accidents caused by drowsiness. One such technology is the Anti-Sleep Alarm (ASA) in cars, which is designed to detect signs of drowsiness in drivers and alert them before they fall asleep at the wheel.

In this project, we have built an Anti-Sleep Alarm (ASA), which will detect if the driver's eyes have been closed for more than 3 seconds, after which it will alter the driver through a piezo buzzer. If the driver fails to wake up even after the buzzer rings for 6 seconds, the car will come to an automatic stop.

LIST AND SPECIFICATIONS OF COMPONENTS:-

1. **Arduino Uno:-**

   Arduino Uno is based on the ATmega328P microcontroller and comes with a variety of input and output pins, which can be programmed to perform various functions. It can be powered using a USB cable or an external power supply, and it can be used to control a variety of electronic components such as LEDs, motors, sensors, and displays. Arduino Uno has 14 digital input/output pins, 6 PWM output pins, 6 analog input pins, and 2 serial communication pins. Arduino Uno is used as microcontroller through which we will build our project.



2. **DC Gear Motor:-**

   Specification:-

   a. Max. speed: 100rpm at 12V
   b. Voltage range: 3V – 12V
   c. Max. load current: 330mA at 12V
   d. Max. torque: 3kg-cm at 12V

   DC Gear Motor is being used to indicate whether the car is moving or stationary.

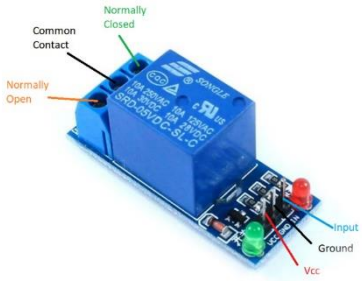
3. **Piezo Buzzer:-**

Specification:-

a. Frequency range: 3300 Hz
b. Voltage range: 3V - 24V
c. Temperature range: 20°C - 60°C

Piezo Buzzer is being used as an alarm to indicate that the driver has fallen asleep.

4. **Single Channel Relay Module:-**



Specification:-

a. Voltage range: 3.75V – 6V
b. Max. Current: 10A
c. Relay active current: 70mA

Single Channel Relay Module is being used to complete or break the connection to the DC Gear Motor, when the driver is awake and asleep respectively.

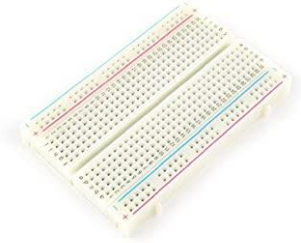
5. **Eye Blink Sensor:-**

Specification:-

a. Voltage rating: 5V
b. TTL Voltage: 0V or 5V

Eye Blink Sensor is being used to detect if the driver has their eyes closed or open.

6. **Breadboard:-**

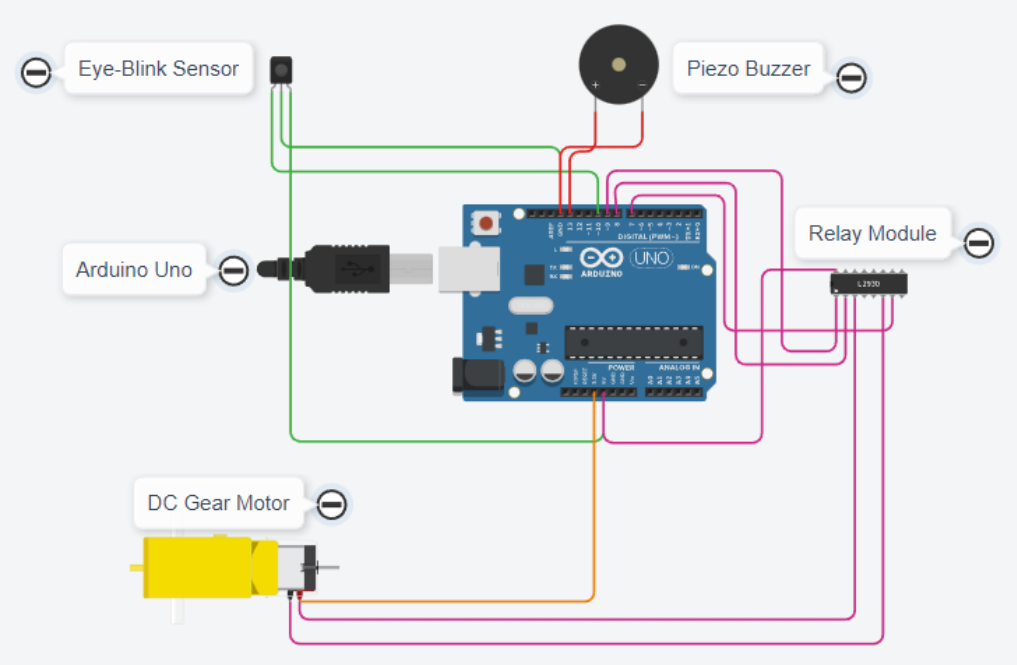


Specification:-

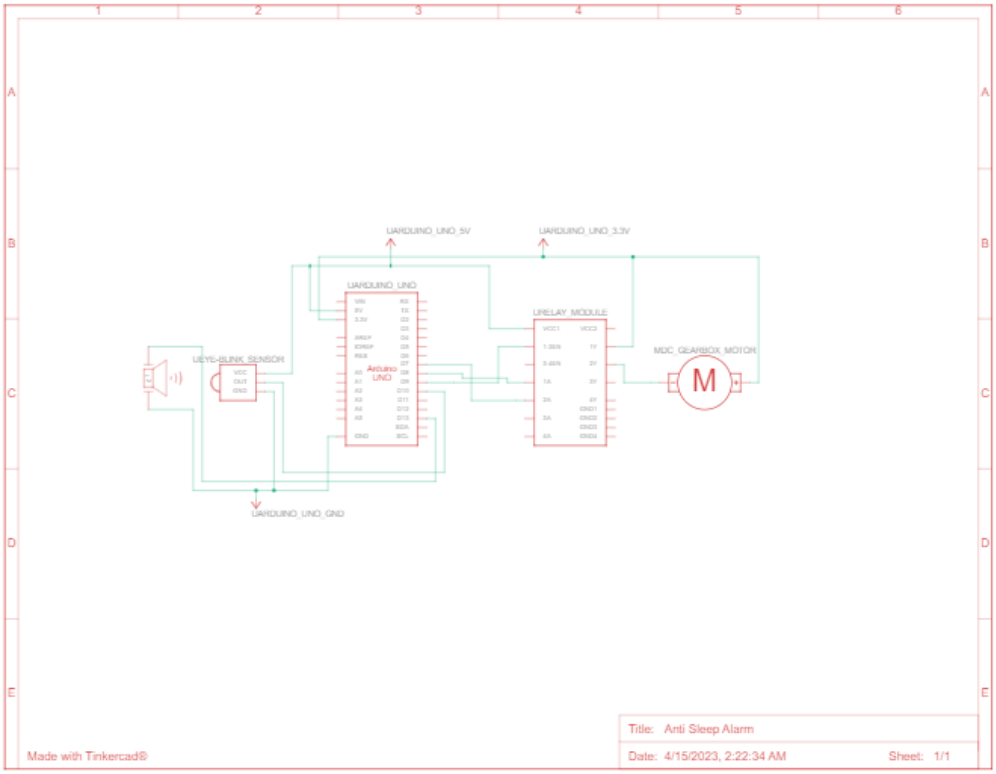
a. Current rating: 5A
b. Dimensions: 6.5*4.4*0.3inches
c. Max. voltage: 1000V(AC)

Breadboard is being used to make connections in our circuit.

CIRCUIT DIAGRAM:-



Circuit Diagram



Schematic Diagram

MODEL DESCRIPTION:-

The Anti Sleep Alarm (ASA) consists of Arduino Uno microcontroller connected to the Relay Module which is connected to the DC Gear Motor. An Eye Blink Sensor is connected which is used to sense the eye movements and gives an indication as to whether the driver is asleep or awake. A Piezo Buzzer is connected to alert the driver if they have gone to sleep to wake them.

CODE:-

```
#define buzzer 13
#define Relay A0
int val=0;
static const int sensorPin = 10;                // sensor input pin
int SensorStatePrevious = HIGH;                 // previous state of the
sensor

unsigned long minSensorDuration = 3000;    // Time we wait before  the
sensor is active as long
unsigned long minSensorDuration2 = 6000;
unsigned long SensorLongMillis;             // Time in ms when the sensor
was active
bool SensorStateLongTime = false;               // True if it is a long active

const int intervalSensor = 50;                  // Time between two readings
sensor state
unsigned long previousSensorMillis;             // Timestamp of the latest
reading

unsigned long SensorOutDuration;                // Time the sensor is
active in ms

//// GENERAL ////

unsigned long currentMillis;          // Variable to store the number of
milliseconds since the Arduino has started
```

```
void setup() {
  Serial.begin(9600);               // Initialise the serial monitor

  pinMode(sensorPin, INPUT);          // set sensorPin as input
  Serial.println("Press button");
  pinMode(Relay,OUTPUT);

pinMode(buzzer,OUTPUT);
}

// Function for reading the sensor state
void readSensorState() {
  // Serial.print(SensorStatePrevious);
  val = analogRead(buzzer);
  Serial.print(val);
  // If the difference in time between the previous reading is larger than
interval sensor
  if(currentMillis - previousSensorMillis > intervalSensor) {

    // Read the digital value of the sensor (LOW/HIGH)
    int SensorState = digitalRead(sensorPin);
    if (SensorState == LOW){
      SensorState = HIGH;
    }
    else
    {
      SensorState = LOW;
    }
    // If the button has been active AND
    // If the sensor wasn't activated before AND
    // IF there was not already a measurement running to determine how
long the sensor has been activated
    if (SensorState == LOW && SensorStatePrevious == HIGH
&& !SensorStateLongTime) {
     SensorLongMillis = currentMillis;
       SensorStatePrevious = LOW;

      Serial.println("Button pressed");
    }
    Serial.print(SensorState);
```

```
 // Calculate how long the sensor has been activated
   SensorOutDuration = currentMillis - SensorLongMillis;

    // If the button is active AND
    // If there is no measurement running to determine how long the
sensor is active AND
    // If the time the sensor has been activated is larger or equal to the
time needed for a long active
    if (SensorState == LOW && !SensorStateLongTime &&
SensorOutDuration >= minSensorDuration) {
      SensorStateLongTime = true;
      digitalWrite(buzzer,HIGH);
      Serial.println("Button long pressed");
    }
    if (SensorState == LOW && SensorStateLongTime &&
SensorOutDuration >= minSensorDuration2) {
     SensorStateLongTime = true;
      digitalWrite(Relay,HIGH);
      delay(1000);
      Serial.print(SensorStatePrevious);
      Serial.println("Button long pressed");
    }

    // If the sensor is released AND
    // If the sensor was activated before
    if (SensorState == HIGH && SensorStatePrevious == LOW) {
      SensorStatePrevious = HIGH;
      Serial.println("Here");
      Serial.print(SensorState);
      SensorStateLongTime = false;
      digitalWrite(Relay,LOW);
      digitalWrite(buzzer,LOW);
      Serial.println("Button released");


    }

    // store the current timestamp in previousSensorMillis
   previousSensorMillis = currentMillis;
  }
}
```

```
void loop() {

  currentMillis = millis();    // store the current time
  readSensorState();           // read the sensor state

}
```

MODEL WORKING:-

1. Supply is given to Arduino Uno through a USB cable.
2. When the driver is in normal/awake condition, when the driver blinks normally, the condition is assumed that a normal human eye blink takes around 50ms to blink. Hence, a normal eye blink would not trigger the piezo buzzer.
3. When the driver falls asleep, a buffer of 3 seconds is given after which the piezo buzzer is triggered to alert the driver.
4. If the driver does not awake 3 sec after the buzzer has been activated, then the relay module cuts the supply to the DC Gear Motor, which will make the car come to a complete stop.

COST ESTIMATE OF PROJECT:-

1. Arduino Uno: Rs. 470
2. Breadboard: Rs. 120
3. Piezo Buzzer: Rs. 35
4. Eye Blink Sensor: Rs. 700
5. DC Gear motor: Rs. 60
6. Relay Module: Rs. 250
7. Jumper wires and cables: Rs. 100

Total cost: Rs. 1735

REFERENCES:-

1. Arduino Uno Documentation
2. TinkerCAD
3. Wikipedia