https://www.pes.edu/wp-content/uploads/2019/09/pes_logo.png

**END SEMESTER ASSESSMENT (ESA) B.TECH. (CSE)**

**IV SEMESTER**

**UE18CS256 – MICROPROCESSOR AND COMPUTER ARCHITECTURE LABORATORY**

**PROJECT REPORT**

**ON**

AUTOMATED CAR USING ARDUINO

SUBMITTED BY

**NAME SRN**

1. **Atharva Gupta PES2UG19CS074**
2. **Atul Anurag PES2UG19CS075**
3. **Bhargav Narayanan P PES2UG19CS088**
4. **Danish Hashmi PES2UG19CS101**

**JANUARY – MAY 2021**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**ELECTRONIC CITY CAMPUS,**

**BENGALURU – 560100, KARNATAKA, INDIA**

|  |  |  |
| --- | --- | --- |
| TABLE OF CONTENTS | | |
| Sl.No | TOPIC | PAGE No |
|  | ABSTRACT OF THE PROJECT | 3 |
|  | CIRCUIT DIAGRAM |  |
|  | ARDUINO CODE |  |
|  | SCREEN SHOTS OF THE OUTPUT |  |
|  | REFERENCES |  |

**ABSTRACT OF THE PROJECT:**

This is the age of automation where human efforts are reducing to a great extent. Making lives simpler and smarter is the aim of Automation. With smartness of Automation comes information and awareness of the technology around us. With the continuous progress and evolution in information technology and the rising demands of safe travel, it has become necessary to find better and innovative systems to aid human life and make it easier. In current day and age, where the COVID-19 Pandemic has affected us all so deeply, we tried making a system where the main propagation technique of covid is avoided, as well as made a couple general safety related equipment.

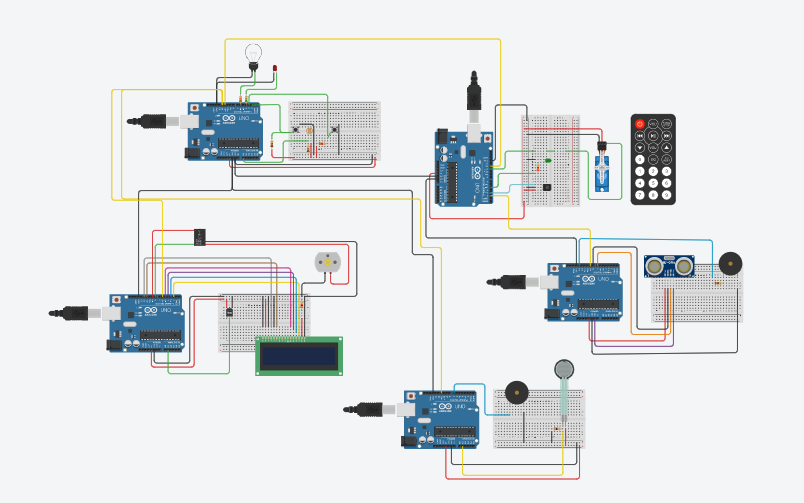
The most common cause of traffic accidents is the Driver error. With cell phones and other electronic media, in-car entertainment systems, the growing traffic, and complicated road systems, this problem has become bigger than ever. So, to reduce impact in the car, we have made 2 general safty related equipment which are the Sensors for airbags deployal as well as a reverse camera for better visibility. Car automation is a technology with the use of which we can control different things or we can keep a track of the vehicle for the security comfort and efficiency. Multiple applications have been developed so as to support the safety and security of the vehicle. There is a wide range of potential social, economic and environmental impacts on the concept of Autonomous driving. Each of these impacts revolves around the inherent efficiency gained by doing a task normally as performed by humans by implementing an optimized computer algorithm to do the task.

The main goal of automation is to provide specific needs with reduced or minimal user intervention.

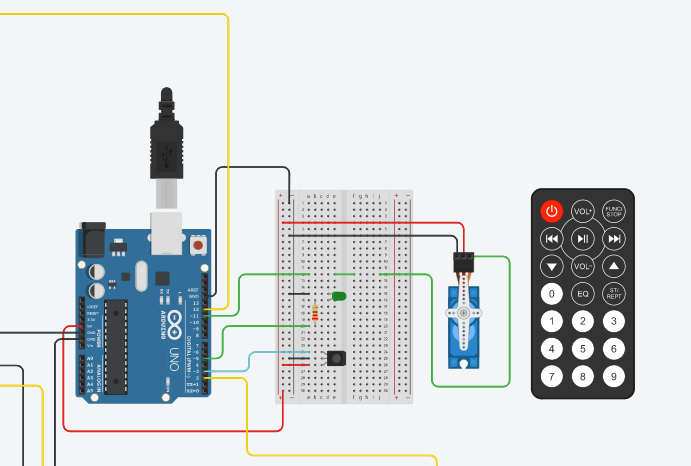
The functionalities of the project include: -

* Utilizing an IR sensor controlled by an IR remote to lock and unlock the door and to power an ultrasonic distance sensor while engaging reverse gear.
* Utilizing a backup sensor for better safety in case of low visibility
* An automatic Light sensor for both the front as well as the rear of the car
* An automatic air-conditioning unit with increases or decreases the temperature based on current conditions in the car
* An airbag sensor for the safety of the passengers of the car incase of an accident

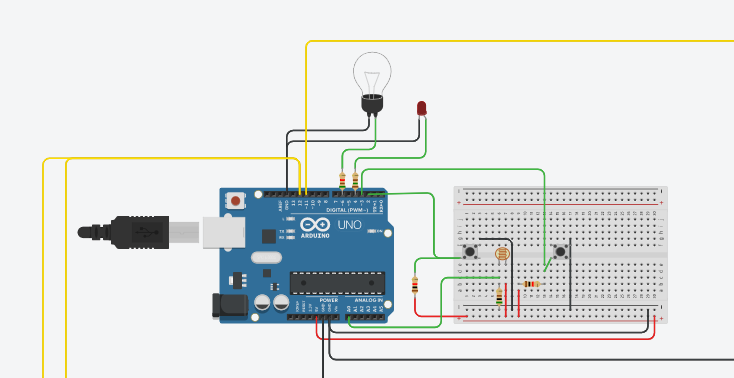
**CIRCUIT DIAGRAM:**



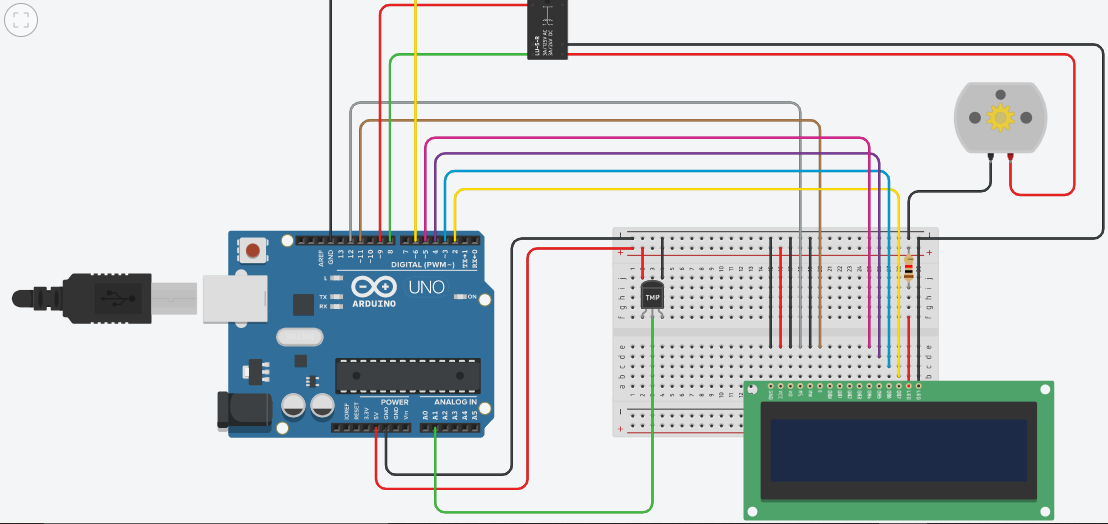
**Arduino 1: IR sensor with remote and Micro Servo**



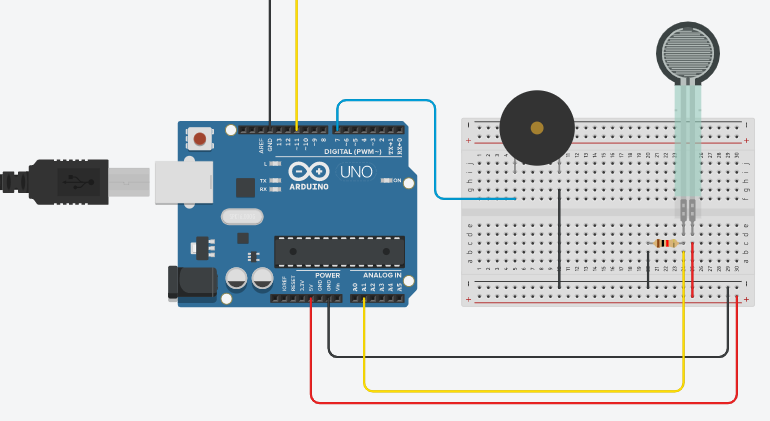
**Arduino 2: LDR with Headlight**



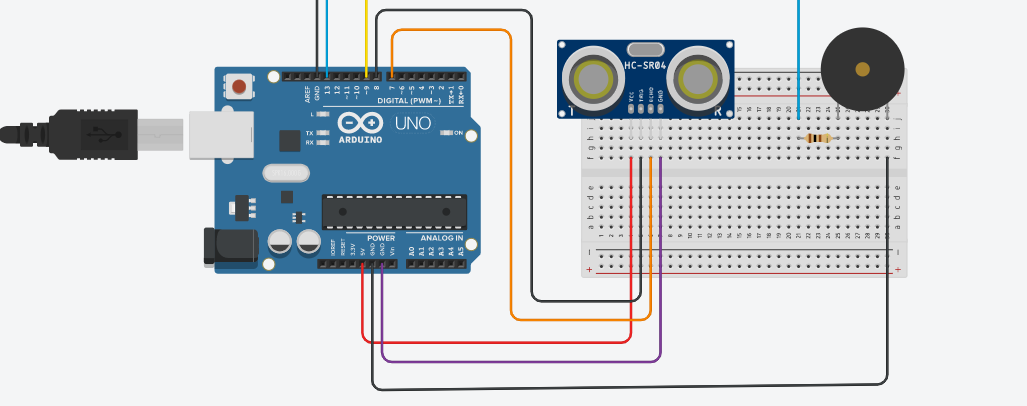
**Arduino 3: Temperature sensor with LCD Display**



**Arduino 4: Force sensor with buzzer**



**Arduino 5: Ultrasonic Distance Sensor with buzzer**



**ARDUINO CODE:**

**Arduino 1:**

#include <IRremote.h>

#include <Servo.h>

int RECV\_PIN = 3;

int signal=0;

int servoPin = 11;

Servo servo;

int pos = 0;

IRrecv irrecv(RECV\_PIN);

decode\_results results;

void setup()

{

Serial.begin(9600);

irrecv.enableIRIn();

pinMode(signal, OUTPUT);

servo.attach(servoPin);

servo.write(0);

}

void loop() {

if (irrecv.decode(&results)) {

switch(results.value)

{

case 16582903: Serial.println("1");

digitalWrite(5,HIGH);

digitalWrite(12,HIGH);

//delay(10);

for (pos; pos < 90; pos += 1)

{

servo.write(pos);

}

break;

case 16615543: Serial.println("2");

digitalWrite(5,LOW);

digitalWrite(12,LOW);

//delay(10);

for (pos; pos > 0; pos -= 1)

{

servo.write(pos);

}

break;

case 16599223: Serial.println("3");

digitalWrite(2,HIGH);

break;

case 16591063: Serial.println("4");

digitalWrite(2,LOW);

break;

default: Serial.println(results.value);

//delay(10);

}

irrecv.resume();

}

}

**Arduino 2:**

#include <avr/io.h>

#include <avr/interrupt.h>

#include <util/delay.h>

#include <LiquidCrystal.h>

//#include <.h>

int engine\_on =0;

int adc\_reading =0,

adc\_val;

int input=11;

int state=0;

void setup()

{

pinMode(input, INPUT);

}

void init\_ports()

{

DDRC &= ~(1<<PC0);

DDRD &= ~((1<<PD2)|(1<<PD3));

DDRD |= ((1<<PD4)|(1<<PD6));

PORTD &= ~(1<<PD4);

PORTD |= (1<<PD6);

}

int read\_adc()

{

ADCSRA |= (1<<ADSC);

while((ADCSRA & 0x10) == 0x10);

{

adc\_reading = (ADCH<<8);

adc\_reading |= ADCL;

}

return (adc\_reading);

}

int main()

{

init\_ports();

// registers-------------

sei();

Serial.begin(9600);

//for engine ON button(PD2) & engine OFF (PD3)

EICRA |= ((1<<ISC11)|(1<<ISC01));

EIMSK |= ((1<<INT0)|(1<<INT1));

// adc

ADMUX |= (1<<REFS0);

ADCSRA |= (1<<ADEN);

//-----pwm--

TCCR0A |= (1<<WGM01);

OCR0A = 200;

OCR0B = 255;

TIMSK0 |= ((1<<OCIE0A)|(1<<OCIE0B));

while(1)

{

state=digitalRead(input);

Serial.println(state);

if(engine\_on == 1 && state==HIGH)

{

PORTD |= (1<<PD4);

adc\_val=read\_adc(); //read adc value

//Serial.println(adc\_val);

if(adc\_val >= 0 && adc\_val < 250 ) //low light,bright headlight

{

TCCR0B |= 0x00;

OCR0A = 240;

OCR0B = 255;

TCCR0B |= ((1<<CS00)|(1<<CS02));

}

else if(adc\_val >= 250 && adc\_val < 500)//medium light,medium brightness

{

TCCR0B |= 0x00;

OCR0A = 100;

OCR0B =255;

TCCR0B |= ((1<<CS00)|(1<<CS02));

}

else if(adc\_val >= 500) //bright light, low headlight

{

TCCR0B |= 0x00;

OCR0A = 20;

OCR0B = 255;

TCCR0B |= ((1<<CS00)|(1<<CS02));

}

}

else

{

PORTD &= ~(1<<PD6);

PORTD &= ~(1<<PD4);

}

}

}

ISR(INT0\_vect)

{

engine\_on =1;

digitalWrite(12, HIGH);

}

ISR(INT1\_vect)

{

engine\_on =0;

digitalWrite(12, LOW);

}

ISR(TIMER0\_COMPA\_vect)

{

PORTD &= ~(1<<PD6);

}

ISR(TIMER0\_COMPB\_vect)

{

PORTD |= (1<<PD6);

}

**Arduino 3:**

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

float temp;

int tempPin = A1;

int relayPin = 8;

int input=6;

int state=0;

#define fan 9

void setup(){

pinMode(input,INPUT);

pinMode(fan, OUTPUT);

pinMode(relayPin, OUTPUT);

here:

state=digitalRead(input);

Serial.println(state);

if(state==HIGH)

{

lcd.begin(16, 3);

lcd.setCursor(2, 0);

lcd.print("Temp Control");

lcd.setCursor(3, 1);

lcd.print("Starting..");

delay(20);

lcd.clear();

}

if(state==LOW)

goto here;

}

void loop()

{

if(state==HIGH)

{

lcd.setCursor(0,2);

temp = analogRead(tempPin);

//temp = temp\*0.48828125;

float voltage = temp \* 5.0;

voltage /= 1024.0;

// print out the voltage

//lcd.print(voltage); lcd.println(" volts");

// now print out the temperature

float temperatureC = (voltage - 0.5) \* 100 ; //converting from 10 mv per degree wit 500 mV offset

//to degrees ((voltage - 500mV) times 100)

lcd.setCursor(0, 0);

lcd.print("Temp = ");

lcd.setCursor(9,0);

lcd.print(temperatureC);

lcd.println(" C");

lcd.setCursor(0,1);

if(temperatureC >= 20)

{

poweronRelay();

if(temperatureC >= 20 && temperatureC <= 25)

{

analogWrite(fan,51);

lcd.print("Fan Speed: 20% ");

delay(20);

lcd.clear();

}

else if(temperatureC <= 35)

{

analogWrite(fan,102);

lcd.print("Fan Speed: 40% ");

delay(20);

lcd.clear();

}

else if(temperatureC <= 40)

{

analogWrite(fan,153);

lcd.print("Fan Speed: 60% ");

delay(20);

lcd.clear();

}

else if(temperatureC <= 44)

{

analogWrite(fan,200);

lcd.print("Fan Speed: 80% ");

delay(20);

lcd.clear();

}

else if(temperatureC >= 45)

{

analogWrite(fan,255);

lcd.print("Fan Speed: 100% ");

delay(20);

lcd.clear();

}

}

else if(temperatureC < 20)

{

poweroffRelay();

lcd.print("Fan Speed: 0% ");

delay(20);

lcd.clear();

}

state=digitalRead(input);

}

if(state==LOW)

{

lcd.clear();

state=digitalRead(input);

}

}

void poweronRelay()

{

digitalWrite(relayPin, HIGH);

}

void poweroffRelay()

{

digitalWrite(relayPin, LOW);

analogWrite(fan,0);

}

**Arduino 4:**

const int buzzer = 7;

const int forcePin = A1;

int ForceValue = 0;

int input=11;

int state=0;

void setup()

{

pinMode(input, INPUT);

pinMode(buzzer, OUTPUT);

pinMode(forcePin, INPUT);

Serial.begin(9600);

}

void loop()

{

state=digitalRead(input);

Serial.println(state);

if(state==HIGH)

{

ForceValue = analogRead(forcePin);

Serial.println(ForceValue);

if (ForceValue >= 300)

tone(7,523,1000);

else

noTone(7);

}

else

noTone(7);

}

**Arduino 5:**

int trig = 8;

int echo = 7;

float reading;

int input=9;

int state=0;

int buzzer=13;

void setup()

{

pinMode(trig, OUTPUT);

pinMode(echo, INPUT);

pinMode(input, INPUT);

pinMode(buzzer,OUTPUT);

Serial.begin(9600);

}

void loop()

{

state=digitalRead(input);

Serial.println(state);

if(state==HIGH)

{

float distance;

digitalWrite(trig,LOW);

delayMicroseconds(2);

digitalWrite(trig,HIGH);

delayMicroseconds(2);

digitalWrite(trig,LOW);

reading = pulseIn(echo,HIGH);

distance=reading\*0.017;

if(distance<50)

{

tone(buzzer,200,100);

delay(10);

noTone(buzzer);

delay(10);

}

Serial.print("The distance is : ");

Serial.print(distance);

Serial.println("cm");

delay(20);

}

}

**SCREEN SHOTS OF THE OUTPUT:**

**REFERENCES**