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**

|  |  |  |
| --- | --- | --- |
| Sl. No. | Topic | Page No. |
| 1 | Abstract of the Project |  |
| 2 | Circuit Diagram |  |
| 3 | Arduino Code |  |
| 4 | Output Screenshots |  |
|  | References |  |

**ABSTRACT OF THE PROJECT:**

This project aims at developing a control system model for modern cars.

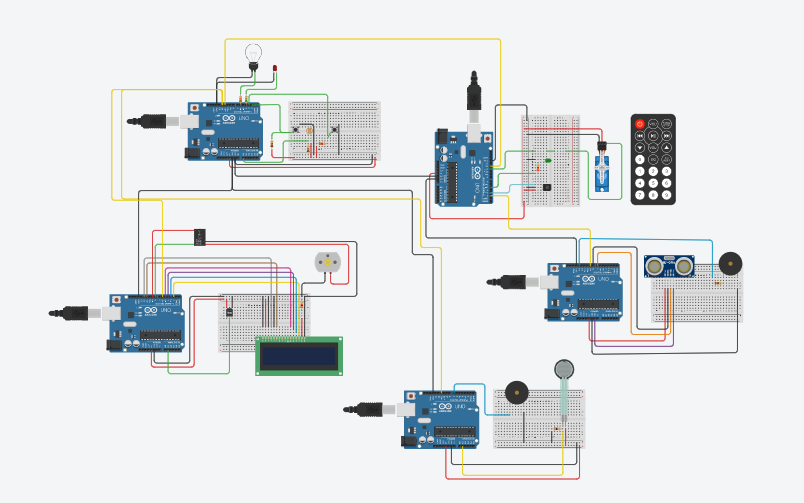
It is built to address few issues in the real world scenario.

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

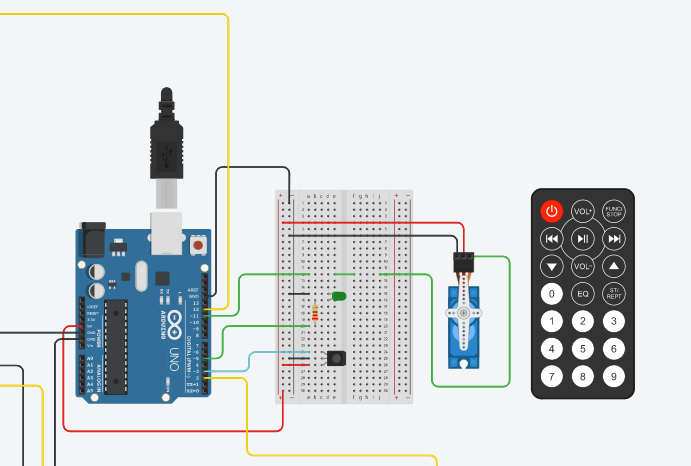
The functionalities of the project include:-

* 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.

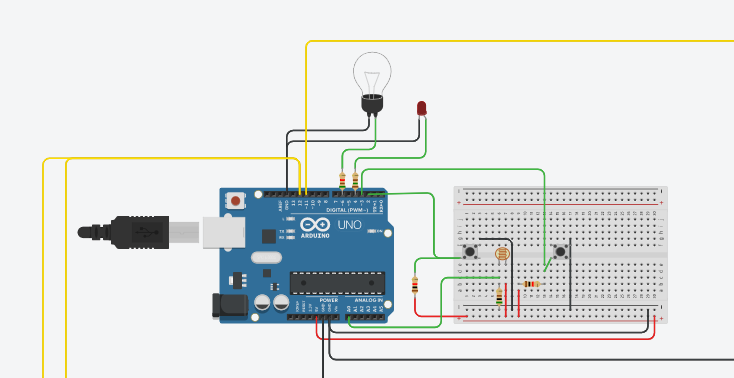
**CIRCUIT DIAGRAM:**



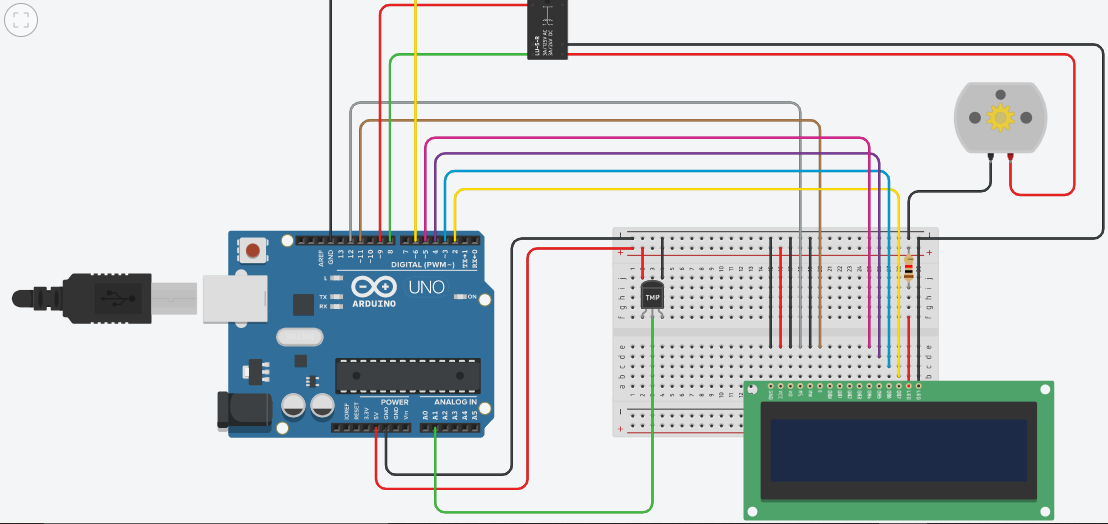
**Arduino 1: IR sensor with remote and Micro Servo to access the car**



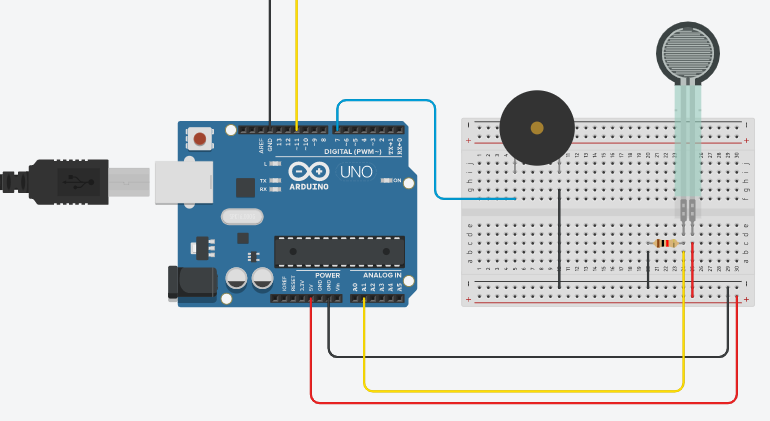
**Arduino 2: Adaptive Lighting system using LDR**



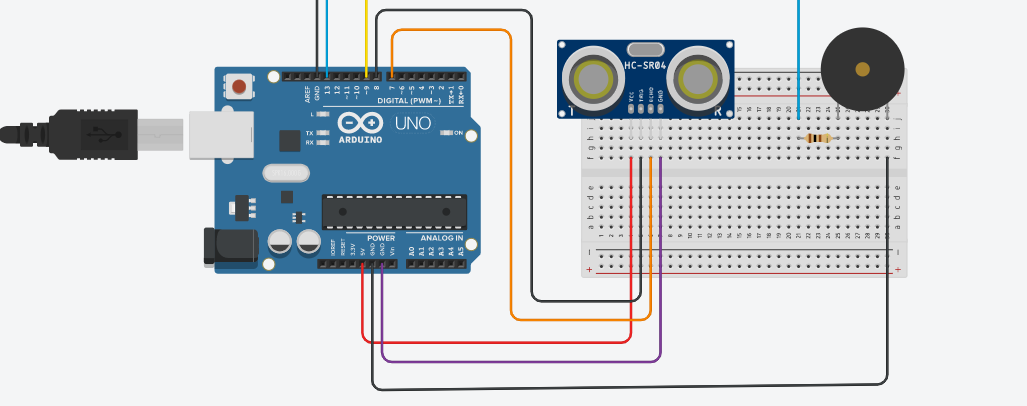
**Arduino 3: Temperature sensor with LCD Display for automatic Air Conditioning System**



**Arduino 4: Force sensor with buzzer for impact detection**



**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**

* **www.tinkercad.com**
* **www.arduino.cc**
* **www.tutorialspoint.com**
* **www.javatpoint.com**