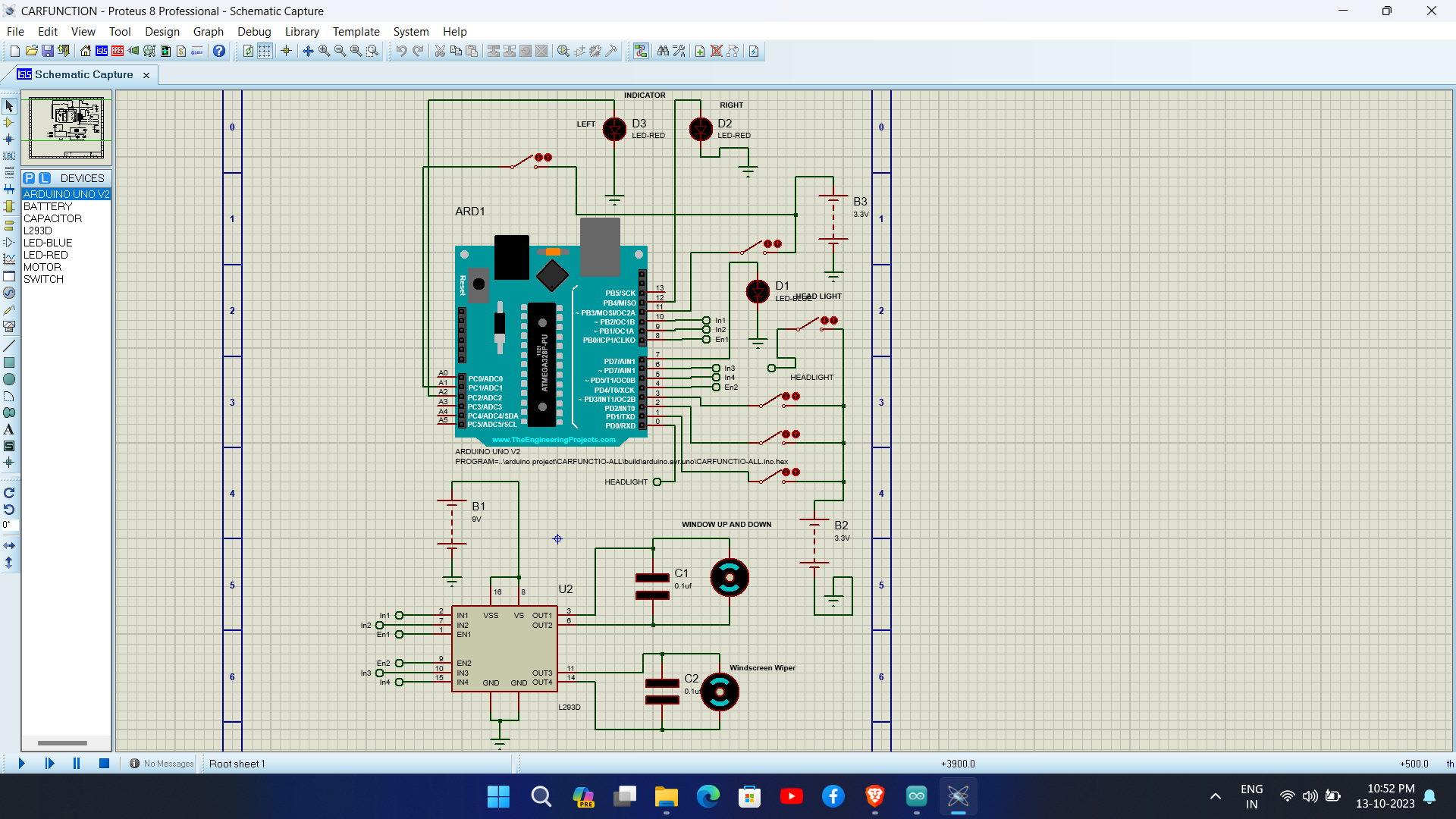
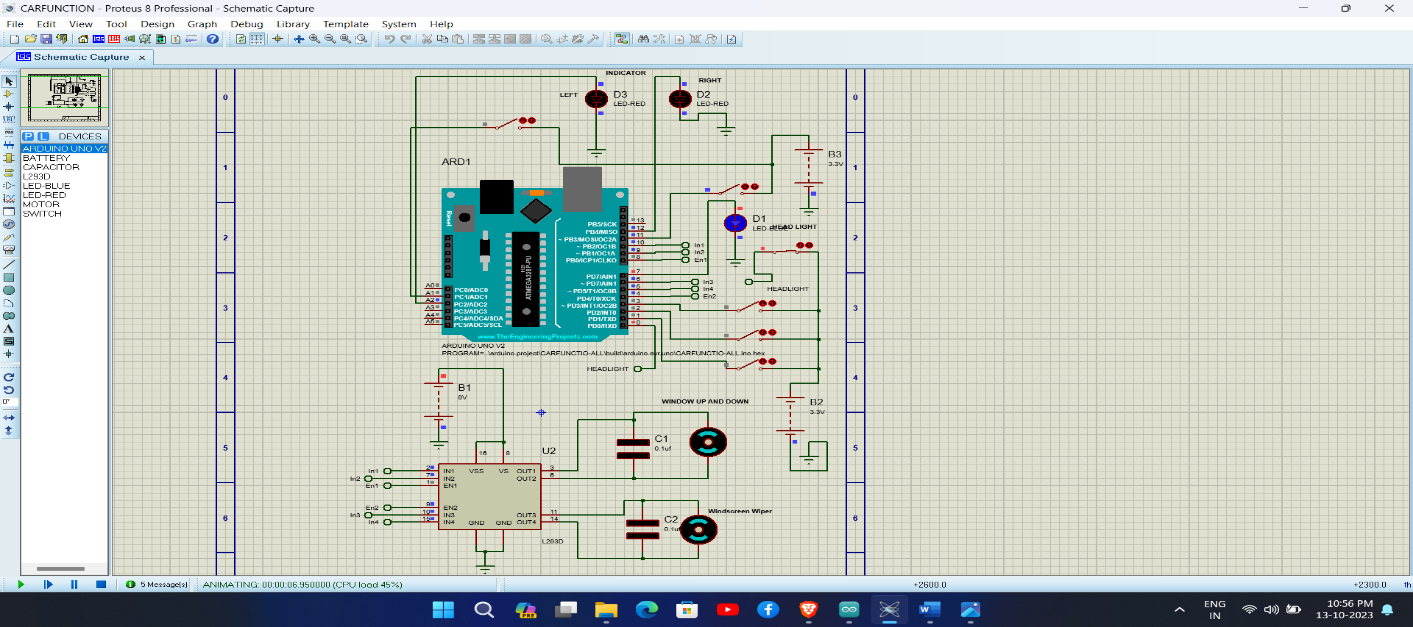
The pins are associated with switches and buttons that dictate actions related to the car's window, wiper, headlights, and indicators. The windowUpSwitchPin and windowDownSwitchPin (pins 1 and 2) regulate the movement of the car window, allowing it to go up or down accordingly. The wiperSwitchPin (pin 3) governs the wiper, initiating the wiper action when pressed. The buttonPin (pin 0) and corresponding ledPin (pin 7) work in harmony to control and indicate the headlight functionality, providing a simulation of headlight activation. Additionally, the buttonPin2 (pin 11) and its associated ledPin2 (pin 12) operate in sync to enable and signify car indicator actions. A similar setup exists with buttonPin3 (pin A1) and ledPin3 (pin A2), providing an additional set of indicator controls and indicators.

Schematic Diagram: -

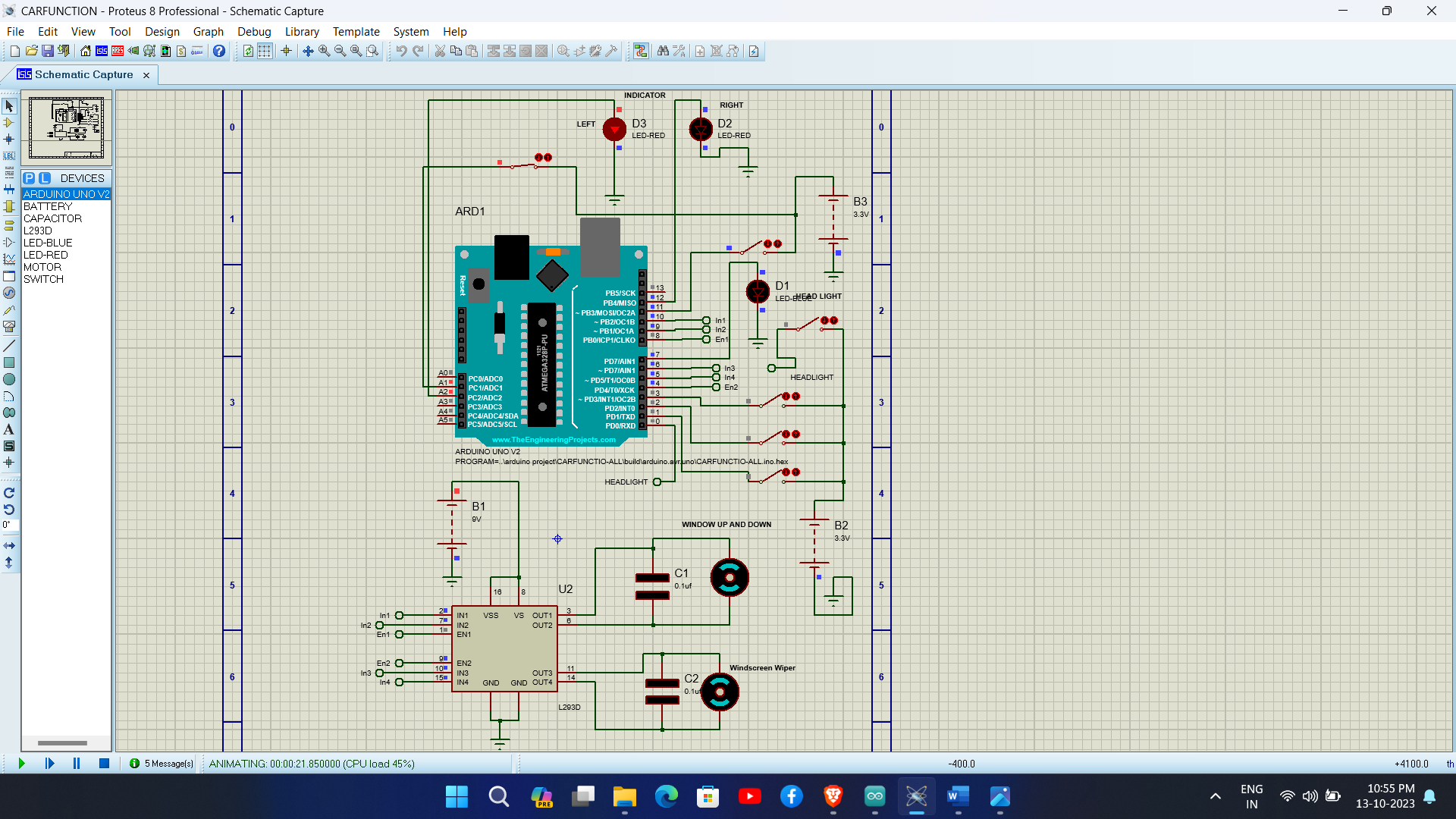
I Have simulated this circuit in proteus 8 professional. Here I am attaching head light and indicator simulation screen shot. Also, I’m mentioning Arduino code. With all work functionalities.

1. Head light simulation result.

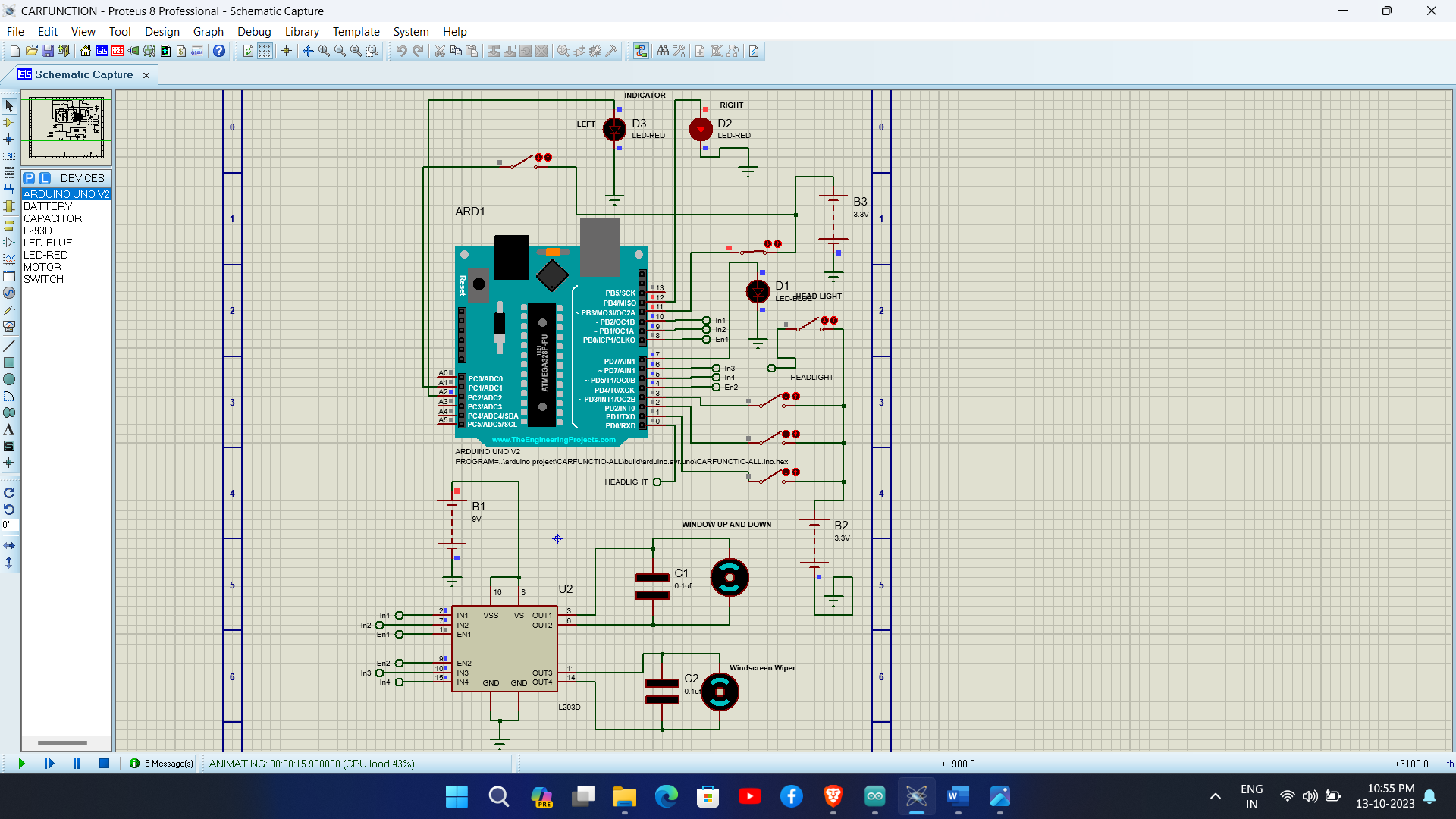


1. Indicator simulation. Left and right indicator

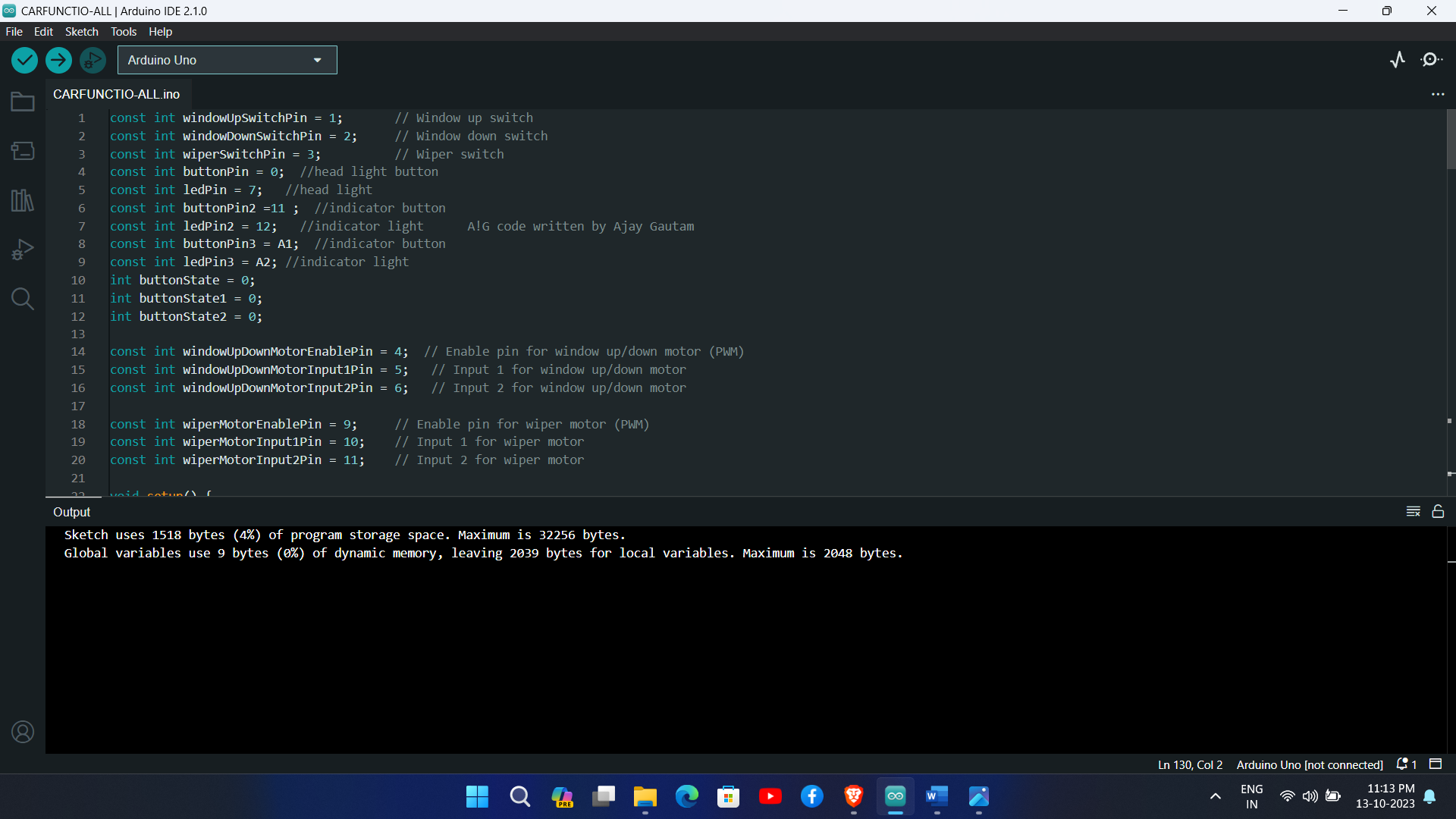








Code: - code is successfully compiled.



const int windowUpSwitchPin = 1;       // Window up switch

const int windowDownSwitchPin = 2;     // Window down switch

const int wiperSwitchPin = 3;          // Wiper switch

const int buttonPin = 0;  //head light button

const int ledPin = 7;   //head light

const int buttonPin2 =11 ;  //indicator button

const int ledPin2 = 12;   //indicator light      A!G code written by Ajay Gautam

const int buttonPin3 = A1;  //indicator button

const int ledPin3 = A2; //indicator light

int buttonState = 0;

int buttonState1 = 0;

int buttonState2 = 0;

const int windowUpDownMotorEnablePin = 4;  // Enable pin for window up/down motor (PWM)

const int windowUpDownMotorInput1Pin = 5;   // Input 1 for window up/down motor

const int windowUpDownMotorInput2Pin = 6;   // Input 2 for window up/down motor

const int wiperMotorEnablePin = 9;     // Enable pin for wiper motor (PWM)

const int wiperMotorInput1Pin = 10;    // Input 1 for wiper motor

const int wiperMotorInput2Pin = 11;    // Input 2 for wiper motor

void setup() {

  pinMode(windowUpSwitchPin, INPUT);

  pinMode(windowDownSwitchPin, INPUT);

  pinMode(wiperSwitchPin, INPUT);

  pinMode(ledPin2, OUTPUT);

  pinMode(buttonPin2, INPUT); // A!G code written by Ajay Gautam

  pinMode(ledPin3, OUTPUT);

  pinMode(buttonPin3, INPUT);

  pinMode(windowUpDownMotorEnablePin, OUTPUT);

  pinMode(windowUpDownMotorInput1Pin, OUTPUT);

  pinMode(windowUpDownMotorInput2Pin, OUTPUT);

  pinMode(wiperMotorEnablePin, OUTPUT);

  pinMode(wiperMotorInput1Pin, OUTPUT);

  pinMode(wiperMotorInput2Pin, OUTPUT);

  pinMode(ledPin, OUTPUT);

  pinMode(buttonPin, INPUT);

  // Initialize motors to stop

  stopWindowUpDownMotor();

  stopWiperMotor();

}

void loop() {

  bool windowUpSwitchState = digitalRead(windowUpSwitchPin) == HIGH;

  bool windowDownSwitchState = digitalRead(windowDownSwitchPin) == HIGH;

  bool wiperSwitchState = digitalRead(wiperSwitchPin) == HIGH;

  // Control window up/down motor

  if (windowUpSwitchState) {

    operateWindowUpMotor();

  } else if (windowDownSwitchState) { // A!G code written by Ajay Gautam

    operateWindowDownMotor();

  } else {

    stopWindowUpDownMotor();

  }

  // Control wiper motor

  if (wiperSwitchState) {

    operateWiperMotor();

  } else {

    stopWiperMotor();

  }

buttonState = digitalRead(buttonPin);

  // A!G code written by Ajay Gautam

  if (buttonState == HIGH) {

    digitalWrite(ledPin, HIGH);

  } else {

    digitalWrite(ledPin, LOW);

  }

  buttonState1 = digitalRead(buttonPin2);

  if (buttonState1 == HIGH) {

    digitalWrite(ledPin2, HIGH);

  } else {

    digitalWrite(ledPin2, LOW);

  }

  buttonState2 = digitalRead(buttonPin3);

  if (buttonState2 == HIGH) {

    digitalWrite(ledPin3, HIGH);

  } else {

    digitalWrite(ledPin3, LOW);

  }

}

void operateWindowUpMotor() {

  digitalWrite(windowUpDownMotorInput1Pin, HIGH);

  digitalWrite(windowUpDownMotorInput2Pin, LOW);

  analogWrite(windowUpDownMotorEnablePin, 255);  // Full speed

}

void operateWindowDownMotor() {

  digitalWrite(windowUpDownMotorInput1Pin, LOW);

  digitalWrite(windowUpDownMotorInput2Pin, HIGH);

  analogWrite(windowUpDownMotorEnablePin, 255);  // Full speed

}

 // A!G code written by Ajay Gautam

void stopWindowUpDownMotor() {

  digitalWrite(windowUpDownMotorInput1Pin, LOW);

  digitalWrite(windowUpDownMotorInput2Pin, LOW);

  analogWrite(windowUpDownMotorEnablePin, 0);  // Stop

}

void operateWiperMotor() {

  // Rotate 90 degrees clockwise

  digitalWrite(wiperMotorInput1Pin, HIGH);

  digitalWrite(wiperMotorInput2Pin, LOW);

  analogWrite(wiperMotorEnablePin, 255);  // Full speed

  delay(5000);  // Rotate for 5 seconds

  // Stop the motor

  stopWiperMotor();

  delay(500);  // Wait for a brief moment

  // Rotate 90 degrees anticlockwise

  digitalWrite(wiperMotorInput1Pin, LOW);

  digitalWrite(wiperMotorInput2Pin, HIGH);

  analogWrite(wiperMotorEnablePin, 255);  // Full speed

  delay(5000);  // Rotate for 5 seconds

  // Stop the motor

  stopWiperMotor();

}

 // A!G code written by Ajay Gautam

void stopWiperMotor() {

  digitalWrite(wiperMotorInput1Pin, LOW);

  digitalWrite(wiperMotorInput2Pin, LOW);

  analogWrite(wiperMotorEnablePin, 0);  // Stop

}

 // A!G code written by Ajay Gautam

The software is crashing while I am doing motor simulation. Because this project has become too much bulky. So I’m sharing my old motor driver project. Please feel free to check.

GitHub link: - <https://github.com/AjayGautam1199/L293d-motor-driver-simple-car-moving-forward-and-backword>.