EST Practical Activity Report

Submitted for

**ENGINEERING DESIGN-II (UTA024)**

# Submitted by:

|  |  |
| --- | --- |
| Abhishek Gupta | 102316027 |
| Satvik Sharma | 102316026 |
| Divjot Singh Gulati | 102316025 |
| Dixant Sharma | 102316022 |
| Kashish rana | 102316021 |
| Swastik | 102316020 |

**BE Second Year Batch – 2P11**

# Submitted to:

**Mr. Divjot Singh**

****

**Computer Science and Engineering Department**

**Thapar Institute of Engineering & Technology, Patiala**

**Jan-June 2024-25 EVENSEM**

**DECLARATION**

We declare that this project report is based on our own work carried out during the course of our study in our Engineering-design II Computer Lab under the supervision of Mr. Narinder Singh.

We assert that the statements made and conclusions drawn are an outcome of our own research work.

We further certify that the work contained in this report is original and has been done by us under the general supervision of our supervisor.

We have followed the guidelines provided by the University in writing this report.

We also declare that this project is the outcome of our own effort, that it has not been submitted to any other university for the award of any degree.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Name of Experiments** | **Page No.** |
| **1** | Introduction to Arduino Microcontroller and other modules used in the lab. |  |
| **2** | Write a program in Arduino to blink a single LED |  |
| **3** | Write an Arduino program to blink multiple LEDs using:  ● While loop,  ● Array,  ● Switch Case,  ● If else,  ● Function. |  |
| **4** | Write an Arduino program to design an odd-even pattern and perform the reverse operation.  For example- Sequence = 1 3 5 2 4--- 4 2 5 3 1 |  |
| **5** | Develop a program to blink five LEDs with five distinct patterns, each implemented in a separate function using analogWrite. Describe each pattern in your code. |  |
| **6** | Introduction to the serial monitor |  |
| **7** | WAP to print the following pattern:  (Use string read function)  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Name-  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Class-  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Roll No.  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |  |
| **8** | Write an Arduino program that takes data from a serial monitor and uses it to control the brightness, number of blinks, and which LED blinks. |  |
| **9** | Write a program for virtual LED Dice using a random function. |  |
| **10** | WAP to perform the following movements of Robo Car (Buggy): forward, backward, left, right, clockwise and anti-clockwise motion. (Use a function for each) |  |

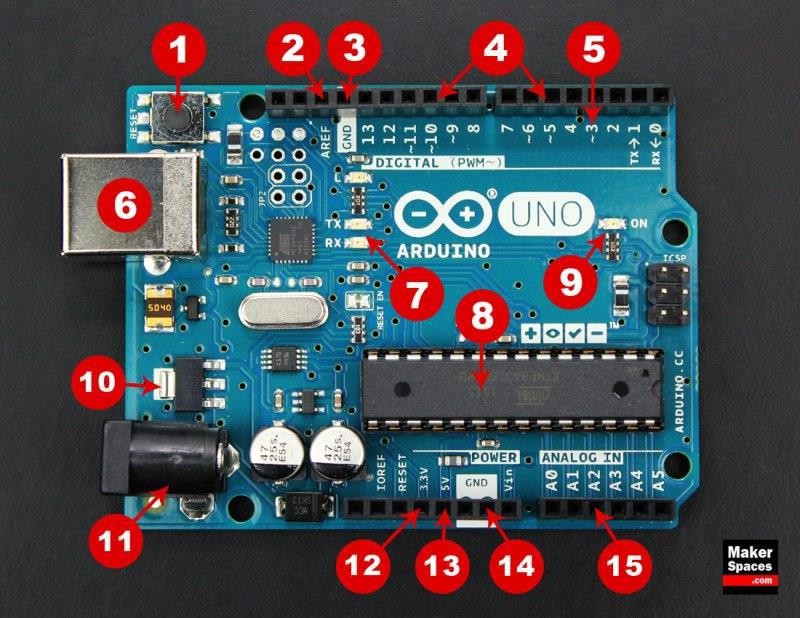
**EXPERIMENT-1**

**OBJECTIVE:** Introduction to Arduino Microcontroller and other modules used in the lab.

**SOFTWARE USED:** Tinkercad Simulator **HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Name of Components** | **Value** |
| **1** | Arduino Uno Micro-Controller | **1** |

**LOGIC/CIRCUIT DIAGRAM:**

****

**Signature of faculty member**

**THEORY:**

The Arduino microcontroller is an open-source electronics platform popular for creating interactive projects. It consists of easy-to-use hardware and software, making it ideal for beginners and professionals alike.

Common Arduino Boards:

1. Arduino Uno: The most popular and widely used board, great for beginners.

2. Arduino Nano: A compact version of the Uno, suitable for smaller projects.

3. Arduino Mega: Offers more digital I/O pins and memory, ideal for more complex projects.

Basic Components of an Arduino Board:

1. Microcontroller: The brain of the board, where the program runs.

2. Digital I/O Pins: Used to read digital signals (HIGH/LOW) or control digital devices like LEDs and buttons.

3. Analog Input Pins: Used to read analog signals from sensors like temperature or light sensors.

4. Power Pins: Provide power to the board and external components.

5. USB Connection: Used to upload programs from the computer to the board and for serial communication.

6. Reset Button: Restarts the program running on the microcontroller.

**DISCUSSIONS**

In this experiment, we get to know about basics of Arduino Uno Microcontroller and its various functions and components.

**Signature of faculty member**

**EXPERIMENT-2**

**OBJECTIVE:** Write a program in Arduino to blink a single LED

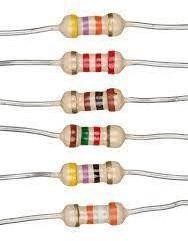
**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

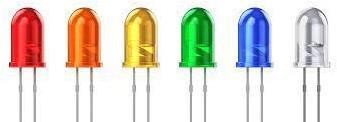
|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 5/NA |
| 4. | LED | 1/NA |
| 5. | Resistor | 1/220 ohm |

**THEORY:**

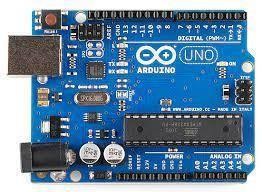
**Resistor:** Resistors are essential components in nearly all electronic circuits and many electrical applications. As their name suggests, they resist the flow of electricity, a function that is crucial for the proper operation of most circuits.



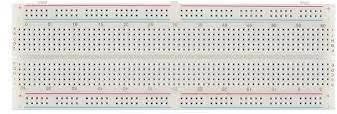
**LED:** A light-emitting diode (LED) is a semiconductor light source that produces light when an electric current passes through it. This occurs as electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons (energy packets).



**Arduino Uno Board:** The Arduino Uno is an open-source microcontroller board featuring the Microchip ATmega328P, developed by Arduino.cc.



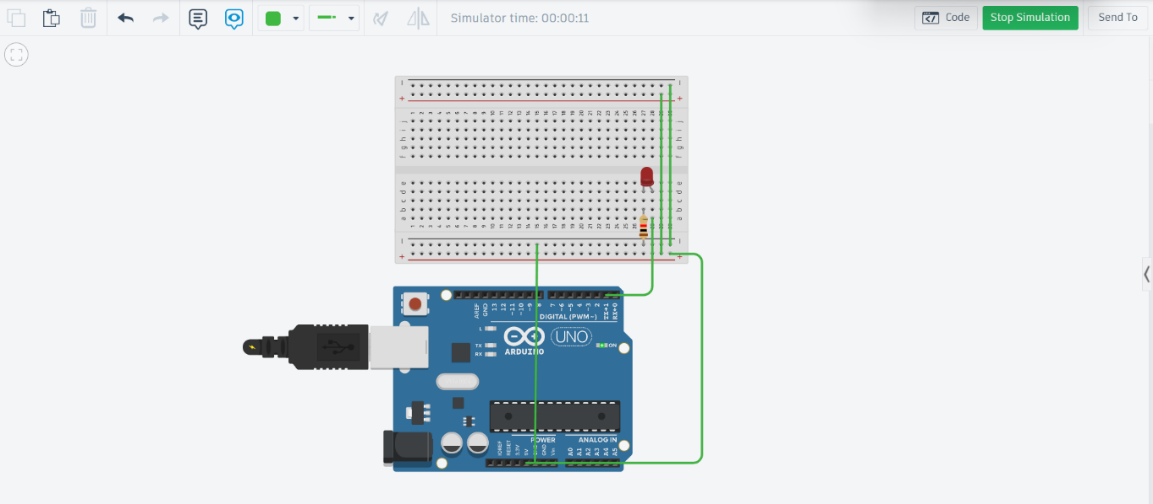
**Breadboard:** A breadboard is used to place components (resistor, capacitor, LED’s etc.) that are wired together. It is used to make temporary circuits.



**Jumper Wires:** A jumper wire is an electric wire that connects remote electric circuits used for printed circuit boards.



**TINKERCAD DIAGRAM:**

****

**CODE:**

void setup() {

pinMode(1,OUTPUT);

}

void loop(){

digitalWrite(1,HIGH);

delay(1000);

digitalWrite(1,LOW);

delay(1000);

**Signature of faculty member**

**EXPERIMENT-3**

**OBJECTIVE:** WAP to blink multiple LEDs using WAP to blink multiple LEDs using

a) While loop

b) Array

c) Switch Case

d) If,else

e) Function

**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 5/NA |
| 5. | Resistor | 5/220 ohm |

**CODE:**

const int arr[] = {1, 2, 3, 4, 5};

const int size = 5;

int ledIndex = 0;

// Function to blink an LED

void blinkLED(int pin) {

digitalWrite(pin, HIGH);

delay(500);

digitalWrite(pin, LOW);

delay(500);

}

void setup() {

for (int i = 0; i < size; i++) {

pinMode(arr[i], OUTPUT);

}

}

void loop() {

// Forward blink using for loop

for (int i = 0; i < size; i++) {

blinkLED(arr[i]);

}

delay(1000);

// Forward blink using while loop

int j = 0;

while (j < size) {

blinkLED(arr[j]);

j++;

}

delay(1000);

// Forward blink using switch-case

switch (ledIndex) {

case 0: case 1: case 2: case 3: case 4:

blinkLED(arr[ledIndex]);

break;

}

ledIndex = (ledIndex + 1) % size; // Move to next LED

delay(500);

// Forward blink using if-else

if (ledIndex < size) {

blinkLED(arr[ledIndex]);

} else {

ledIndex = 0;

}

delay(500);

}

**Signature of faculty member**

**EXPERIMENT-4**

**OBJECTIVE:** Write an Arduino program to design an odd-even pattern and perform the reverse operation. For example- Sequence = 1 3 5 2 4--- 4 2 5 3 1

**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 5/NA |
| 5. | Resistor | 5/220 ohm |

**CODE:**

const int arr[] = {1, 2, 3, 4, 5};

const int size = 5;

void setup() {

for (int i = 0; i < size; i++) {

pinMode(arr[i], OUTPUT);

}

}

void loop() {

for (int i = 0; i < size; i += 2) {

digitalWrite(arr[i], HIGH);

}

delay(500);

for (int i = 0; i < size; i += 2) {

digitalWrite(arr[i], LOW);

}

delay(500);

for (int i = 1; i < size; i += 2) {

digitalWrite(arr[i], HIGH);

}

delay(500);

for (int i = 1; i < size; i += 2) {

digitalWrite(arr[i], LOW);

}

delay(500);

delay(1000);

for (int i = 1; i < size; i += 2) {

digitalWrite(arr[i], HIGH);

}

delay(500);

for (int i = 1; i < size; i += 2) {

digitalWrite(arr[i], LOW);

}

delay(500);

for (int i = 0; i < size; i += 2) {

digitalWrite(arr[i], HIGH);

}

delay(500);

for (int i = 0; i < size; i += 2) {

digitalWrite(arr[i], LOW);

}

delay(500);

delay(1000);

}

**Signature of faculty member**

**EXPERIMENT-5**

**OBJECTIVE:** Develop a program to blink five LEDs with five distinct patterns, each implemented in a separate function using analogWrite. Describe each pattern in your code.

**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 4/NA |
| 5. | Resistor | 1/220 ohm |

**CODE:**

const int leds[] = {3, 5, 6, 9, 10}; // LED pins

const int size = 5;

void setup() {

for (int i = 0; i < size; i++) {

pinMode(leds[i], OUTPUT);

}

}

void pattern1() {

// Gradual fade-in and fade-out for all LEDs

for (int brightness = 0; brightness <= 255; brightness += 5) {

for (int i = 0; i < size; i++) {

analogWrite(leds[i], brightness);

}

delay(30);

}

for (int brightness = 255; brightness >= 0; brightness -= 5) {

for (int i = 0; i < size; i++) {

analogWrite(leds[i], brightness);

}

delay(30);

}

}

void pattern2() {

// Chasing effect from left to right

for (int i = 0; i < size; i++) {

analogWrite(leds[i], 255);

delay(200);

analogWrite(leds[i], 0);

}

}

void pattern3() {

// Alternate blinking between odd and even LEDs

for (int i = 0; i < size; i += 2) {

analogWrite(leds[i], 255);

}

delay(500);

for (int i = 0; i < size; i += 2) {

analogWrite(leds[i], 0);

}

delay(500);

for (int i = 1; i < size; i += 2) {

analogWrite(leds[i], 255);

}

delay(500);

for (int i = 1; i < size; i += 2) {

analogWrite(leds[i], 0);

}

delay(500);

}

void pattern4() {

// Increasing brightness one LED at a time

for (int i = 0; i < size; i++) {

for (int brightness = 0; brightness <= 255; brightness += 5) {

analogWrite(leds[i], brightness);

delay(10);

}

delay(200);

}

}

void pattern5() {

// Random blinking of LEDs

for (int i = 0; i < 10; i++) {

int randomLED = leds[random(0, size)];

analogWrite(randomLED, 255);

delay(200);

analogWrite(randomLED, 0);

}

}

void loop() {

pattern1();

delay(1000);

pattern2();

delay(1000);

pattern3();

delay(1000);

pattern4();

delay(1000);

pattern5();

delay(1000);

}

**Signature of faculty member**

# EXPERIMENT-6

**OBJECTIVE:** Introduction to the Serial Monitor in Arduino.

**SOFTWARE USED:** Tinkercad Simulator

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| Sr. No | **Name of Components** | **Value** |
| 1 | Arduino Uno Micro-Controller | **1** |

**THEORY:**

The Serial Monitor is a built-in feature in the Arduino IDE and simulators like Tinkercad, used to send and receive data between the Arduino board and a computer. It is especially useful for debugging and displaying outputs from the Arduino in real-time.

Purpose of Serial Monitor:

1. Allows text-based communication between the Arduino and the user.
2. Helps in displaying sensor values, debugging code, and tracking program execution.
3. Enables sending commands to the Arduino during runtime.

Key Functions Related to Serial Monitor:

1. Serial.begin(baudRate): Initializes serial communication at the specified baud rate (e.g.,Serial.begin(9600)).
2. Serial.print(): Prints data to the Serial Monitor without a newline.
3. Serial.println(): Prints data followed by a newline.
4. Serial.read(): Reads incoming serial data.

Working Principle:

The USB cable connected between the computer and Arduino board allows data transmission. Once Serial.begin() is called in the setup() function, the Arduino is ready to communicate using the Serial Monitor.

**DISCUSSIONS**

In this experiment, we get to know about the use of the Serial Monitor, how it helps in debugging, and its role in real-time data communication with the Arduino board.

**Signature of faculty member**

**EXPERIMENT-7**

**OBJECTIVE:**

WAP to print the following pattern:

(Use string read function)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name-

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Class-

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Roll No.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 5/NA |
| 5. | Resistor | 5/220 ohm |

**CODE:**

String name, className, rollNo;

void setup() {

Serial.begin(9600);

Serial.println("Enter Name: ");

while (Serial.available() == 0);

name = Serial.readString();

Serial.println("Enter Class: ");

while (Serial.available() == 0);

className = Serial.readString();

Serial.println("Enter Roll No.: ");

while (Serial.available() == 0);

rollNo = Serial.readString();

Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Serial.print("Name- ");

Serial.println(name);

Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Serial.print("Class- ");

Serial.println(className);

Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Serial.print("Roll No.- ");

Serial.println(rollNo);

Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

void loop() {

}

**EXPERIMENT-8**

**OBJECTIVE:**

Write an Arduino program that takes data from a serial monitor and uses it to control the brightness, number of blinks, and which LED blinks.

**SOFTWARE USED:** Tinkercad Simulator.

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 5/NA |
| 5. | Resistor | 5/220 ohm |

**CODE:**

const int ledPins[] = {3, 5, 6}; // Define multiple LED pins

int selectedLED = 3; // Default LED

int brightness = 128; // Default brightness (PWM)

int blinkCount = 5; // Default blink count

void setup() {

Serial.begin(9600); // Start serial communication

for (int i = 0; i < 3; i++) {

pinMode(ledPins[i], OUTPUT);

}

}

void loop() {

if (Serial.available()) {

String input = Serial.readStringUntil('\n'); // Read user input

parseInput(input);

controlLED();

}

}

// Function to process input

void parseInput(String input) {

int firstComma = input.indexOf(',');

int secondComma = input.indexOf(',', firstComma + 1);

if (firstComma != -1 && secondComma != -1) {

selectedLED = ledPins[input.substring(0, firstComma).toInt()];

brightness = input.substring(firstComma + 1, secondComma).toInt();

blinkCount = input.substring(secondComma + 1).toInt();

}

}

// Function to control LED based on user input

void controlLED() {

for (int i = 0; i < blinkCount; i++) {

analogWrite(selectedLED, brightness); // Set brightness

delay(500);

analogWrite(selectedLED, 0); // Turn off

delay(500);

    }

}

**Signature of faculty member**

**EXPERIMENT-9**

**OBJECTIVE:** .

Write a program for virtual LED Dice using a random function. **SOFTWARE USED:** Ardiuno IDE

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Arduino Uno Board | 1/NA |
| 2. | Breadboard | 1/NA |
| 3. | Jumper Wires | 9/NA |
| 4. | LED | 5/NA |
| 5. | Resistor | 5/220 ohm |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| 1. | Buggy kit | 1/NA |
| 2. | USB-Cable | 1/NA |

**CODE:**

**const int leds[] = {3, 4, 5, 6, 7, 8};**

**const int size = 6;**

**const int buttonPin = 2;**

**void setup() {**

**Serial.begin(9600);**

**pinMode(buttonPin, INPUT\_PULLUP);**

**for (int i = 0; i < size; i++) {**

**pinMode(leds[i], OUTPUT);**

**}**

**}**

**void rollDice() {**

**int roll = random(1, 7);**

**Serial.print("Dice Roll: ");**

**Serial.println(roll);**

**for (int i = 0; i < size; i++) {**

**digitalWrite(leds[i], LOW);**

**}**

**for (int i = 0; i < roll; i++) {**

**digitalWrite(leds[i], HIGH);**

**}**

**delay(1000);**

**}**

**void loop() {**

**if (digitalRead(buttonPin) == LOW) {**

**rollDice();**

**delay(500);**

**}**

**}**

**Signature of faculty member**

**EXPERIMENT-10**

**OBJECTIVE:**

WAP to perform the following movements of Robo Car (Buggy):

forward, backward, left, right, clockwise and anti-clockwise motion.

(Use a function for each)

**SOFTWARE USED: Ardiuno IDE**

**HARDWARE USED:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Name of the Component** | **QTY/Value** |
| **1.** | **Buggy kit** | **1/NA** |
| **2.** | **USB-Cable** | **1/NA** |

**CODE:**

**const int pins[4]={5,6,7,8};**

**void setup() {**

**for(int i = 0; i < 4; i++){**

**pinMode(pins[i],OUTPUT);**

**}**

**}**

**void forward(){**

**digitalWrite(pins[0],HIGH);**

**digitalWrite(pins[1],LOW);**

**digitalWrite(pins[3],HIGH);**

**digitalWrite(pins[2],LOW);**

**}**

**void backward(){**

**digitalWrite(pins[0],LOW);**

**digitalWrite(pins[1],HIGH);**

**digitalWrite(pins[2],HIGH);**

**digitalWrite(pins[3],LOW);**

**}**

**void left(){**

**digitalWrite(pins[0],HIGH);**

**digitalWrite(pins[1],LOW);**

**digitalWrite(pins[3],LOW);**

**digitalWrite(pins[2],LOW);**

**}**

**void right(){**

**digitalWrite(pins[0],LOW);**

**digitalWrite(pins[1],LOW);**

**digitalWrite(pins[3],HIGH);**

**digitalWrite(pins[2],LOW);**

**}**

**void clockwise(){**

**digitalWrite(pins[0],LOW);**

**digitalWrite(pins[1],HIGH);**

**digitalWrite(pins[3],HIGH);**

**digitalWrite(pins[2],LOW);**

**}**

**void anticlockwise(){**

**digitalWrite(pins[0],HIGH);**

**digitalWrite(pins[1],LOW);**

**digitalWrite(pins[2],HIGH);**

**digitalWrite(pins[3],LOW);**

**}**

**}**

**void loop() {**

**forward();**

**delay(1000);**

**right();**

**delay(1000);**

**left();**

**delay(1000);**

**backward();**

**delay(1000);**

**clockwise();**

**delay(1000);**

**anticlockwise();**

**delay(1000);**

**}**

**Signature of faculty member**