Experiment 4

***Code1***

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

while (!Serial);

Serial.println("Input 1 to Turn LED on and 2 to off");

}

void loop() {

if (Serial.available())

{

int state = Serial.parseInt();

if (state == 1)

{

digitalWrite(13, HIGH);

Serial.println("Command completed LED turned ON");

}

if (state == 2)

{

digitalWrite(13, LOW);

Serial.println("Command completed LED turned OFF");

}

}

}

**Code2**

Graphical user interface

Description automatically generated with medium confidence

void setup()

{

pinMode(2, OUTPUT);

Serial.begin(9600);

while (!Serial);

Serial.println("Input 1 to Turn LED on and 2 to off");

}

void loop()

{

if (Serial.available())

{

int state = Serial.parseInt();

if (state == 1)

{

digitalWrite(2, HIGH);

Serial.println("Command completed LED turned ON");

}

if (state == 2)

{

digitalWrite(2, LOW);

Serial.println("Command completed LED turned OFF");

}

}

}

***Code3***

Diagram

Description automatically generated

int sensorValue = 0;

void setup()

{

pinMode(A0, INPUT);

Serial.begin(9600);

pinMode(9, OUTPUT);

}

void loop()

{

// read the value from the sensor

sensorValue = analogRead(A0);

// print the sensor reading so you know its range

Serial.println(sensorValue);

analogWrite(9, map(sensorValue, 0, 1023, 0, 255));

delay(100); // Wait for 100 millisecond(s)

}

***Code4***

Diagram

Description automatically generated

Servo servo\_1;

void setup()

{

servo\_1.attach(1, 500, 2500);

}

void loop()

{

servo\_1.write(0);

delay(1000); // Wait for 1000 millisecond(s)

servo\_1.write(45);

delay(1000); // Wait for 1000 millisecond(s)

}

***Code5***

A picture containing text, electronics

Description automatically generated

void setup()

{

pinMode(A0, INPUT);

pinMode(9, OUTPUT);

Serial.begin(9600);

}

void loop()

{

int sensorReading = analogRead(A0);

Serial.println(sensorReading);

int modifiedSensorReading = map(sensorReading,26,923,0,256);

analogWrite(9,modifiedSensorReading);

delay(10);

}

***Code6***

Graphical user interface, diagram

Description automatically generated

#include<Servo.h>

int buttonState = 0;

Servo myservo;

int val;

void setup()

{

pinMode(2, INPUT);

pinMode(13, OUTPUT);

myservo.attach(10);

}

void loop()

{

int buttonState = digitalRead(2);

if (buttonState == HIGH) {

val=180;

} else {

val=0;

}

myservo.write(val);

delay(10);

}

***Code7***

Graphical user interface, diagram

Description automatically generated

#include<Servo.h>

Servo myservo;

int sensorValue = 0;

void setup()

{

pinMode(A0, INPUT);

pinMode(13, OUTPUT);

myservo.attach(10);

}

void loop()

{

delay(100);

sensorValue = analogRead(A0);

myservo.write(sensorValue);

delay(10);

}

***Code8***

A picture containing text, electronics

Description automatically generated

#include <Servo.h>

int pos = 0;

Servo servo\_9;

void setup()

{

servo\_9.attach(9, 500, 2500);

}

void loop()

{

for (pos = 0; pos <= 180; pos += 1) {

servo\_9.write(pos);

delay(10); // Wait for 10 millisecond(s)

}

for (pos = 180; pos >= 0; pos -= 1) {

servo\_9.write(pos);

delay(10); // Wait for 10 millisecond(s)

}

}

***Code9***

Diagram

Description automatically generated

#include<Servo.h>

int buttonState = 0;

Servo myservo;

int val;

void setup()

{

pinMode(2, INPUT);

pinMode(10, OUTPUT);

myservo.attach(8);

}

void loop()

{

int buttonState = digitalRead(2);

if (buttonState == HIGH) {

tone(10,250);

val=90;// door opening at 90 degree

delay(10);

noTone(10);

}

else {

val=0;

}

myservo.write(val);

delay(10);

}

***Code10***

Graphical user interface, diagram

Description automatically generated

#include<Servo.h>

Servo myservo;

void setup()

{

pinMode(6, OUTPUT);// green

pinMode(12, OUTPUT);// red

myservo.attach(2);

}

void loop()

{

digitalWrite(12, HIGH);

myservo.write(0);

delay(5000);

digitalWrite(12, LOW);

delay(500);

digitalWrite(6, HIGH);

myservo.write(90);

delay(5000);

digitalWrite(6, LOW);

delay(10);

}