Vedant Sanap 47 Ansh Sarfare 49 Jai Talreja 59 D15A

WT Exp 5

Aim :-

To run a simulation of the project on TinkerCAD.

Theory :-

Tinkercad is a free, web-based 3D modeling and design application that has gained significant popularity among beginners, educators, and hobbyists. Developed by Autodesk, Tinkercad simplifies the process of 3D design, making it accessible to users of all skill levels. Since its launch in 2011, it has become a go-to platform for creating models for 3D printing, electronics prototyping, and coding.

Overview of Tinkercad

Type of Site : 3D modeling, Computer Aided Design (CAD)

Owner : Autodesk

URL : www.tinkercad.com

Commercial : Yes, but the software is free to use

Registration : Required to access full features

Launched : 2011

Written in : WebGL, JavaScript

Key Features and Uses

3D Design and Printing :

Tinkercad is best known for its 3D design capabilities. Users can create detailed 3D models by combining basic shapes such as cubes, spheres, and cylinders. The intuitive interface allows users to drag and drop these shapes onto a work plane and manipulate them to build complex designs.

Electronics Prototyping :

In addition to 3D design, Tinkercad supports electronics prototyping. Users can create and simulate circuits using a variety of components, including microcontrollers, sensors, and LEDs.

Coding :

Tinkercad also includes a block-based coding environment, which is ideal for beginners learning to code.

User-Friendly Interface :

The simplicity of the interface makes it an excellent choice for students, educators, and hobbyists who are new to 3D modeling and design

Accessibility :

Since Tinkercad is a browser-based application, it can be accessed from any device with an internet connection. This makes it highly versatile and convenient for users who need to work on their designs from different locations

Educational Applications :

Tinkercad has become a popular tool in educational settings. It is used by teachers to introduce students to the concepts of 3D design, electronics, and coding.

Historical Context :

The website was launched in 2011 as a web-based 3D modeling tool for WebGL-enabled browsers. By 2012, the company had moved its headquarters to San Francisco, and over 100,000 3D designs had been published by users. In May 2013, Autodesk acquired Tinkercad.

Community and Support :

Tinkercad has a vibrant community of users who share their designs, tutorials, and projects. The platform also offers a range of resources, including video tutorials, forums, and a library of pre-made designs, to help users get started and improve their skills

Tinkercad continues to evolve, offering a powerful yet accessible platform for 3D design, electronics, and coding, making it a valuable tool for both personal and educational use.

Code :-

```
#include <Servo.h>

// Pin definitions
const int trigPin = 7;
const int echoPin = 8;
const int servoPin = 9;
const int moisturePin = 2; // Digital pin for soil moisture sensor // Analog pin for soil moisture sensor
const int moistureThreshold = 500; // Adjust this based on calibration for wet/dry detection

Servo myServo;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
```

```
  myServo.attach(servoPin);
  myServo.write(100);
  delay(1000);
}

void loop() {
  long duration = measureDistance();
  long distance = duration * 0.0344 / 2;

  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.print(" cm");

  if (distance < 20) {
    delay(2000);
    bool isWet = digitalRead(moisturePin); // Read digital value (HIGH = Wet, LOW = Dry)
    // Digital sensor directly gives wet/dry status // Wet if above threshold, else dry

    Serial.print(", Moisture: ");
    Serial.print(isWet);
    Serial.print(" -> ");
    Serial.println(isWet ? "Wet" : "Dry");

    if (isWet) {
      Serial.println("Wet object detected! Moving clockwise.");
      myServo.write(180);
    } else {
      Serial.println("Dry object detected! Moving counterclockwise.");
      myServo.write(20);
    }
    delay(1500);
    Serial.println("Returning to 100 degrees.");
    myServo.write(100);
    delay(2000);
  } else {
    Serial.println(" - No object detected. Servo remains at 100 degrees.");
  }
  delay(2000);
}

long measureDistance() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
```

```
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  return pulseIn(echoPin, HIGH);
}
```

Practical :-

Steps

1.) Connection of components with the Arduino
    a.) Connecting all +5v pins to Arduino's 5v supply using breadboard
    b.) Connecting most GND pins using breadboard (Ultrasonic sensors GND pin directly connected to Arduino board)
    c.) Ultrasonic's TRIG pin connected to D7, ECHO to D8
    d.) Servo's Signal pin to ~9 (D9 also used for PWM, ~ means PWM)

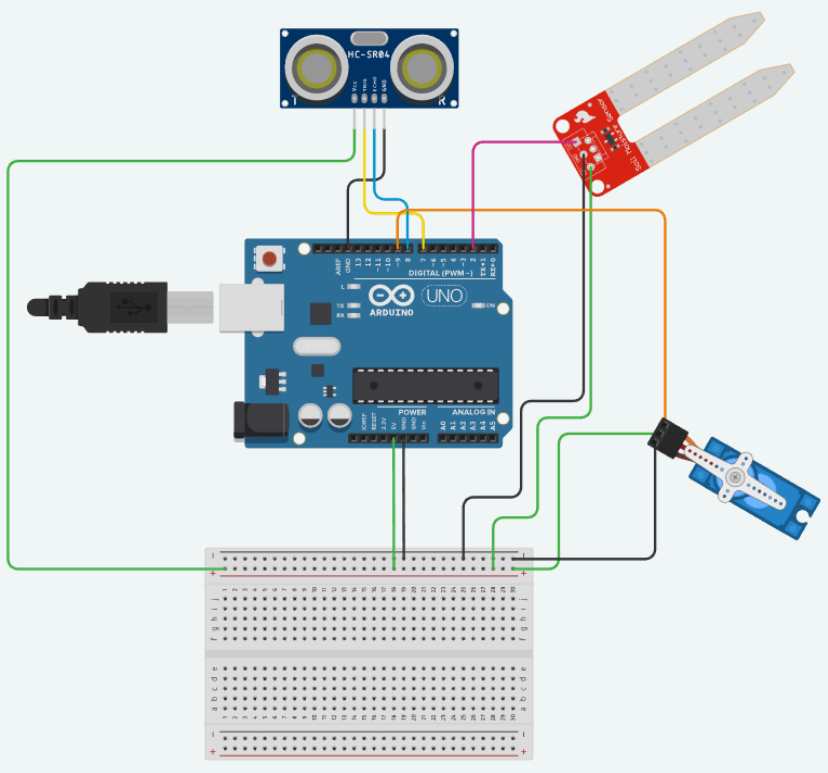
2.) Writing code for the functioning of the Arduino with the components
    a.) Initial code only Ultrasonic and Servo
    b.) Then Ultrasonic, Soil Moisture and Servo
    c.) Then we refined the timings and thresholds for detecting object, and moisture

3.) Using Serial Monitor to see output of the Arduino
    a.) We added logs in the code for easy debugging of the project via the Serial Monitor

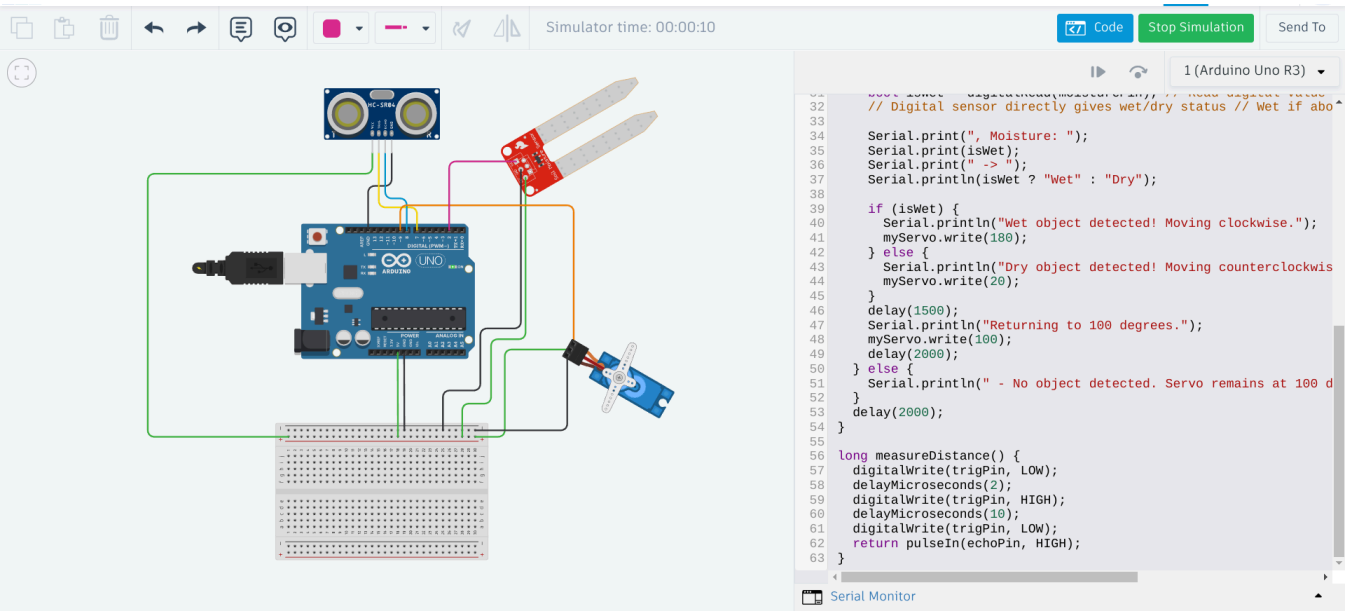
[Circuit diagram and schematic on next page]

Vedant Sanap 47 Ansh Sarfare 49 Jai Talreja 59 D15A

Created circuit diagram

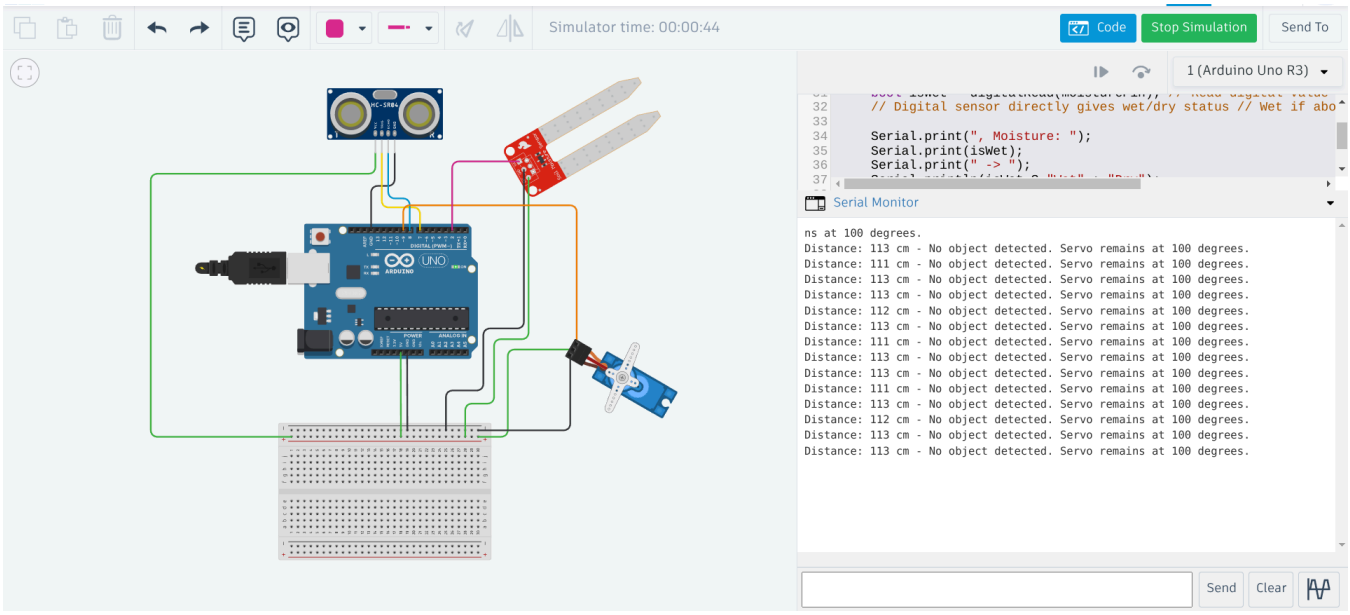


Pasted code and ran simulation, simulation running successfully

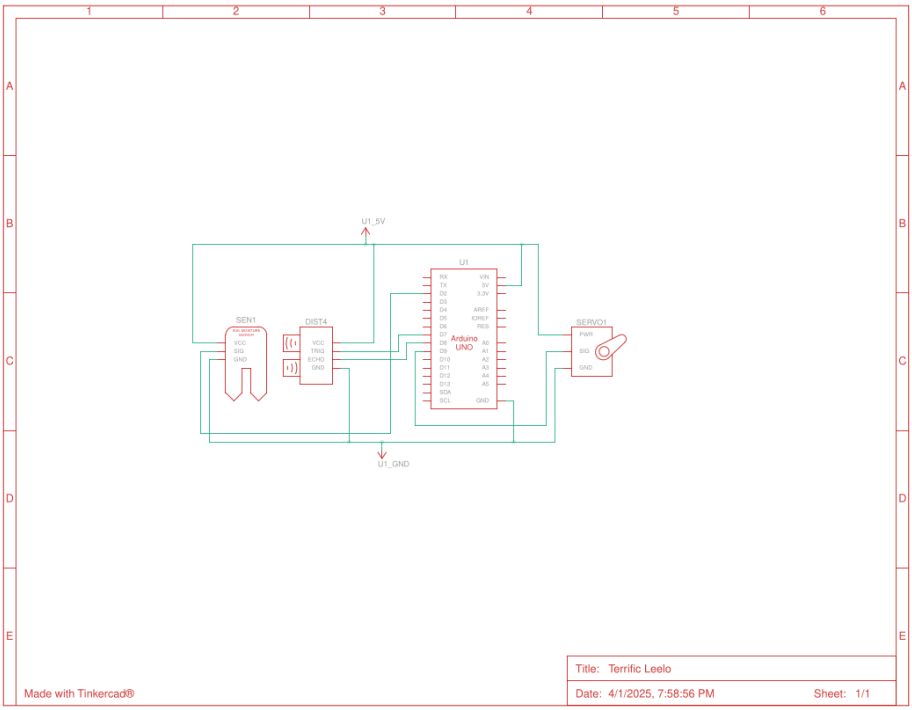


Vedant Sanap 47 Ansh Sarfare 49 Jai Talreja 59 D15A

Serial monitor displays output

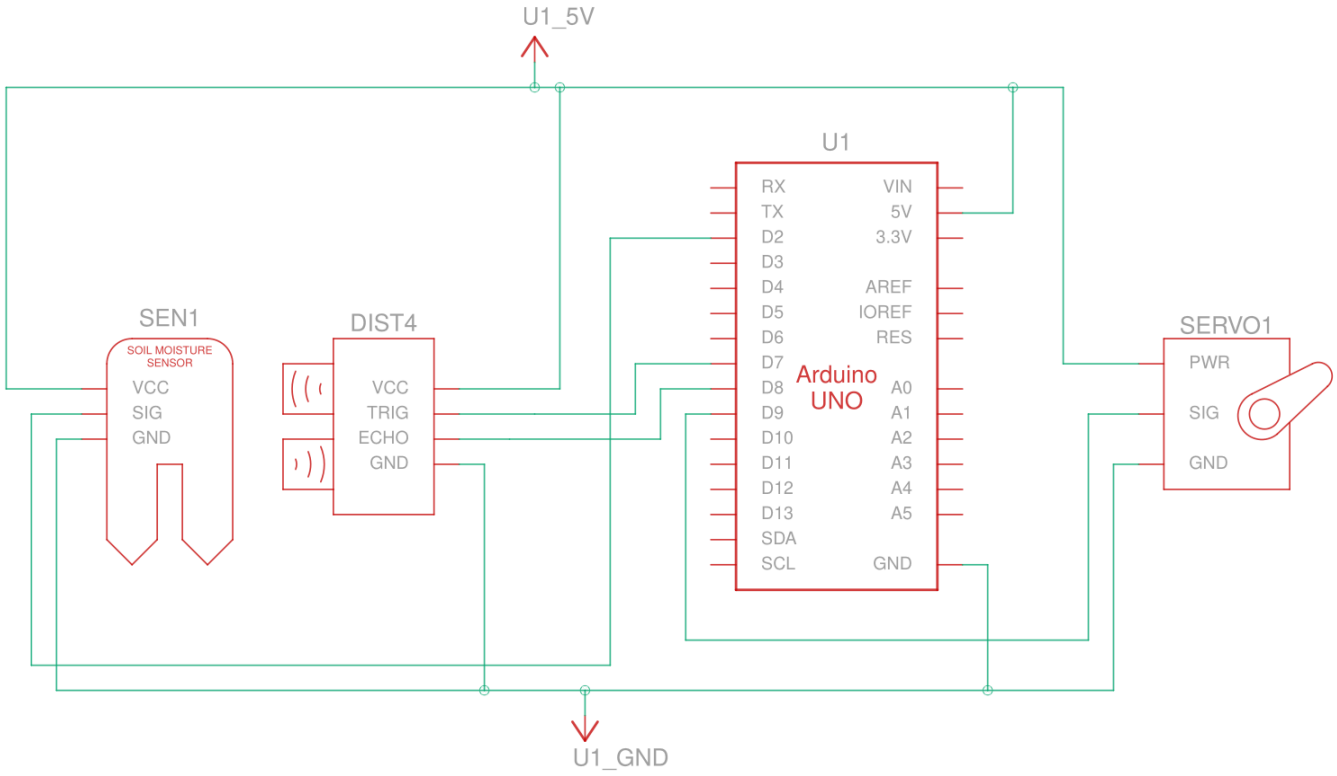


Schematic obtained from TinkerCAD



Vedant Sanap 47 Ansh Sarfare 49 Jai Talreja 59 D15A

Zoomed schematic



Conclusion :-

We successfully ran the simulation of our project, a dry and wet object segregator, using TinkerCAD, and our knowledge of Arduino, how it is supposed to be interfaced and how its code is supposed to be written.