



**Autonomous Irrigation System**

|  |  |
| --- | --- |
| **Name** | **ID** |
| Ashraf Adel | 196280 |
| Farah Aymen | 194233 |
| Mohammed Adel | 197908 |
| Salma Mahmoud | 202235 |

Table of Contents

[1. Schematic Circuit 3](#_Toc73914573)

[2. Components 4](#_Toc73914574)

[3. Code 5](#_Toc73914575)

[4. Link for the video 8](#_Toc73914576)

[5. Hardware Circuit 9](#_Toc73914577)

[6. Explanation 10](#_Toc73914578)

[A. Soil Moisture Sensor 10](#_Toc73914579)

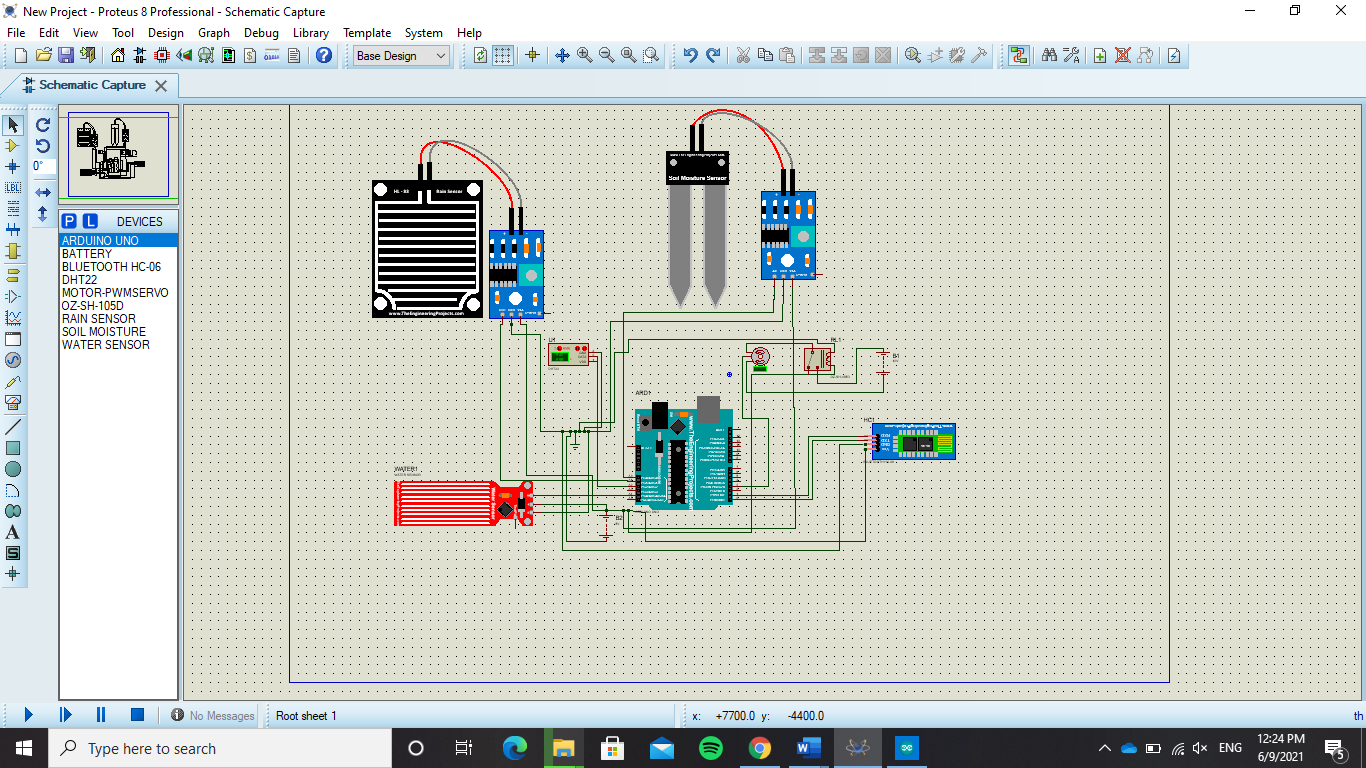
[B. Temperature and Humidity Sensor 10](#_Toc73914580)

[C. Water Level Sensor 11](#_Toc73914581)

[7. Difficulties Faced 12](#_Toc73914582)

[8. References 13](#_Toc73914583)

# Schematic Circuit

The schematic circuit was drawn used proteus. One of the difficulties faced were: the water pump component was unavailable, so it was replaced with a servo motor to represent it, and it was not possible to connect the ground and the Vcc of the components to the Arduino directly so it was substituted with a ground and 5V battery for elaboration.

# Components

1. Temperature and Humidity Sensor, Model DHT 22.
2. Soil moisture Sensor.
3. Bluetooth module, Model HC-05.
4. Water pump.
5. Water Level Sensor.
6. Relay kit.
7. Rain Drop Sensor.

# Code

#include "DHT.h" //Library for humidity sensor

//define pins

#define soilMoisture A0

#define rainSensor A1

#define humiditySensor A2

#define waterSensor A4

int relay = 3;

int moisture; //variable to store value read from moisture sensor

int rain; //variable to store value read from rain sensor

int humidity; //variable to store value read from humidity sensor

int waterLevel; //variable to store value read from water Level sensor

void setup() {

Serial.begin(9600);

DDRD = 0b00001000; //Only relay ->3 is output

PORTD |= (1 << 3); //set relay to be turned off by default

}

void loop() {

moisture = analogRead(soilMoisture); //read value from mositure sensor

rain = analogRead(rainSensor); //read value from rain Sensor

humidity = analogRead(humiditySensor); //read value from humidity sensor

waterLevel = analogRead(waterSensor); //read value from water Level sensor

//condition to check if the tank is empty or not

if (waterLevel < 100)

{

Serial.println("Tank is Empty");

}

//condition to check if Tank is almost empty

else if (waterLevel < 200)

{

Serial.println("Warning! Tank is almost empty...");

}

//condition to check if it's raining

else if (rain < 100 )

{

Serial.println("it's now raining. No need to water the plant.");

}

else

{

//check if plant is too dry

if (moisture > 750)

{

pumpWater(5000); //call function that activates the relay to open the water pump

}

//check if plant in range that needs to be watered

else if (moisture < 750 && moisture > 500 )

{

//check if humidity is not too high

if (humidity > 50)

{

pumpWater(2000);

}

//checks if humidity is high

else

{

pumpWater(4000);

}

}

}

delay(2000);

}

//function that activates the relay to open the water pump and deactiviates it aghain

void pumpWater(int milliSeconds)

{

PORTD |= (1 << 3); // 00000000 | 00001000 == 00001000

delay(milliSeconds);

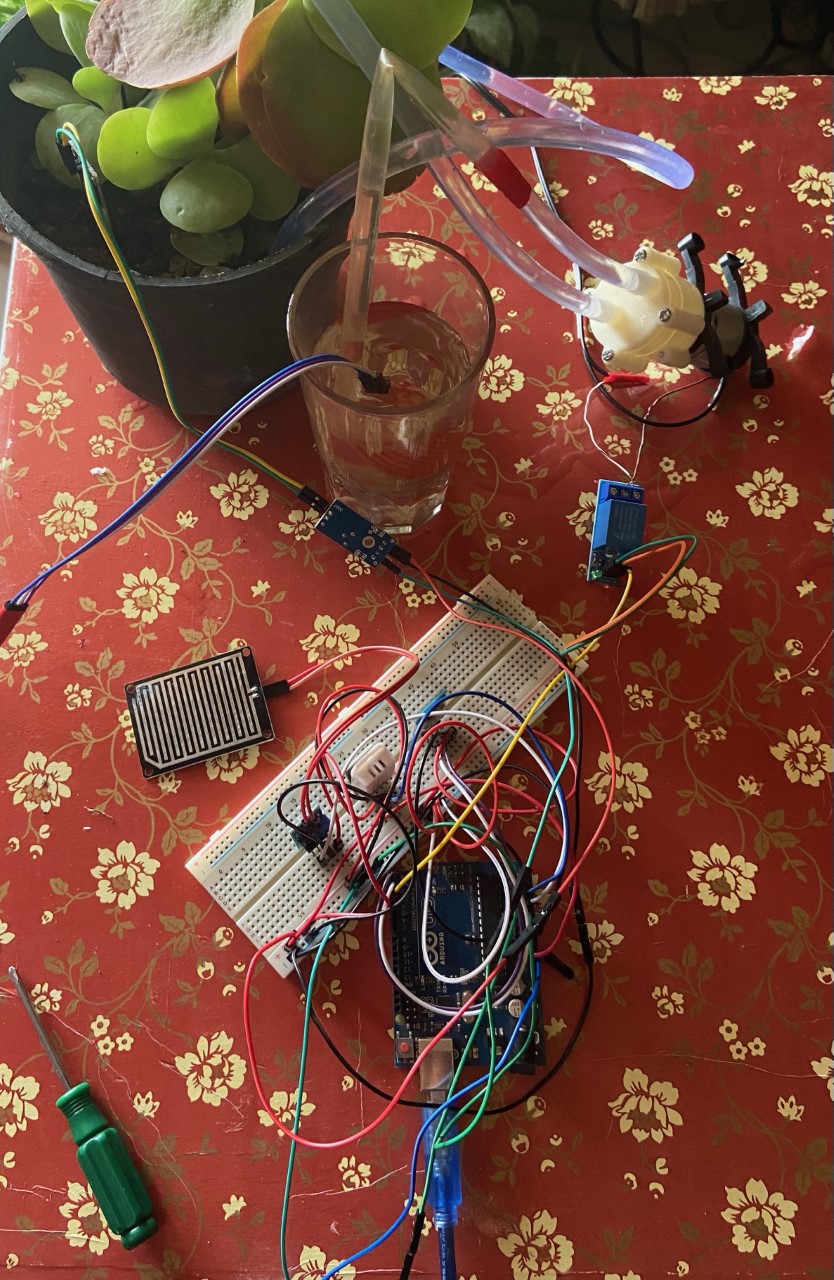
PORTD &= ~(1 << 3); //00000000 & 11110111 == 00000000 \*/

}

# Link for the video

https://drive.google.com/file/d/1thI0mmedi4z7IMYh\_UTH6HAAJbp3EluM/view?usp=sharing

# Hardware Circuit

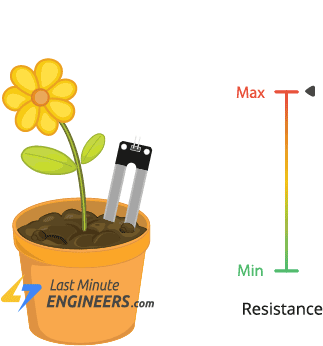


# Explanation

The whole system will be connected on the 5V pin. The system supposedly will test the water level, humidity, and moisture of the soil and at certain values it will detect when it will need to be watered, so it will activate the relay to activate the water pump to water the plant and send a notification to the phone to notify the user and keep them up to date.

## Soil Moisture Sensor

First, the soil moisture sensor is dipped in the soil so it can measure the moisture of the plant and at a certain value when the moisture is low, it should open the replay that will activate the water pump so it can water the plant for enough time and deactivate it afterwards. According to calibration values: <500 is too wet, 500-750 is the target range and >750 is too dry so it needs to be watered [1]. The sensor works as some kind of resistance, as moisture decreases the resistance increases so the voltage sent to the Arduino is low and vice versa (inversely proportional relationship).

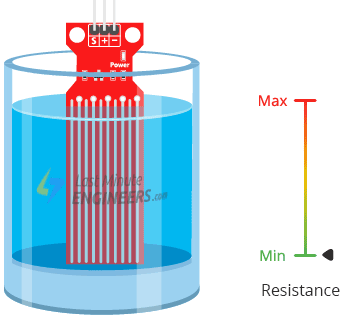


## Temperature and Humidity Sensor

The temperature and humidity sensor used is model DHT-22. This sensor should measure the humidity of air around the plant, as it is rather important for the plant’s biologic processes. Humidity around the plant saturates the leaves with enough water vapour. As humidity level goes higher, water cannot evaporate from the plant and absorb minerals from the soil. While humidity is low and temperature is warm, the plant loses great part of water as it evaporates [2]. So, as humidity decreases the water will lose water fast so the sensor should indicate when the humidity went low so it means that the plant needs more water than usual; as it has lost a lot.

## Water Level Sensor

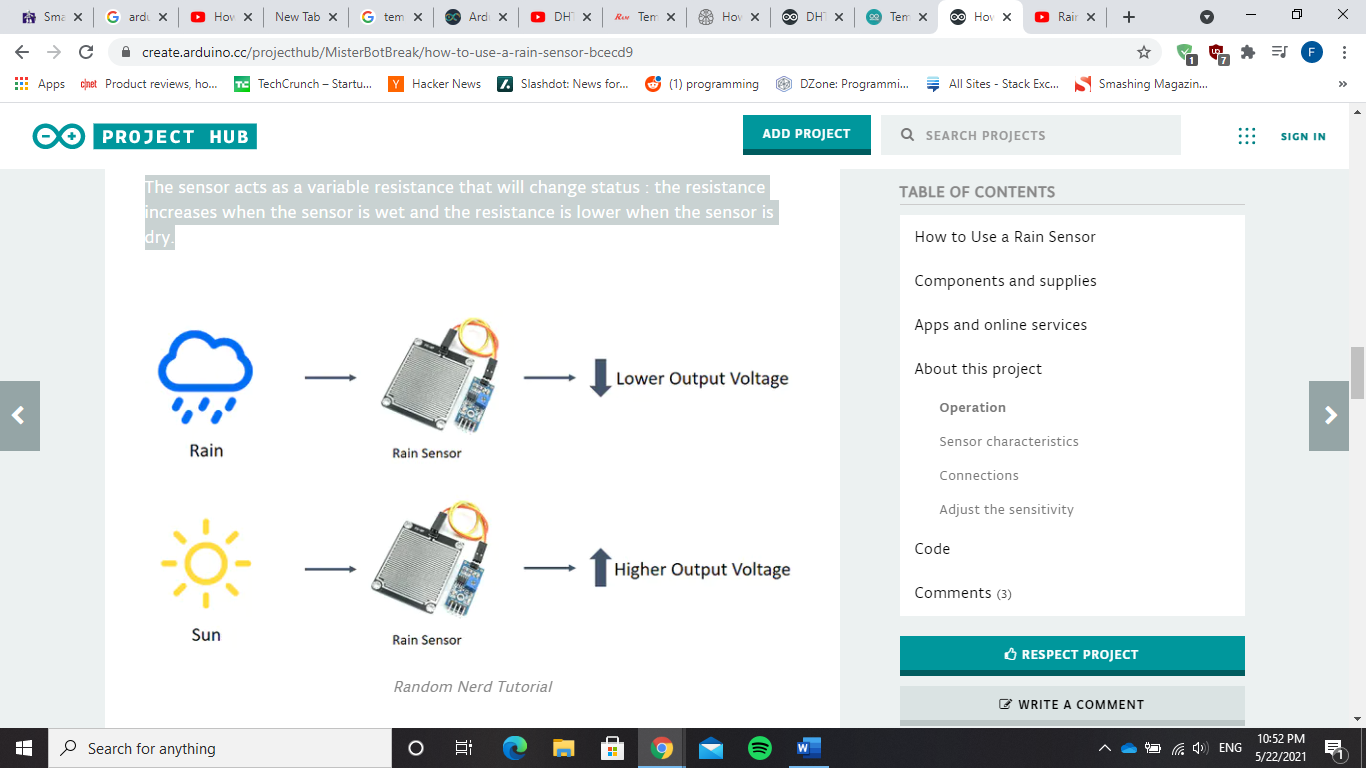
The water level sensor’s purpose that it will measure the water level in the tank that has the water stored in it to water the plant and when it reaches to a low level it notifies the user to refill the tank. The sensor is also some kind of a resistance that the more its surface is covered with water the less resistance there is so it sends high voltage to the Arduino indicating that the tank is filled with water and vice versa (inversely proportional relationship).



## Bluetooth Module

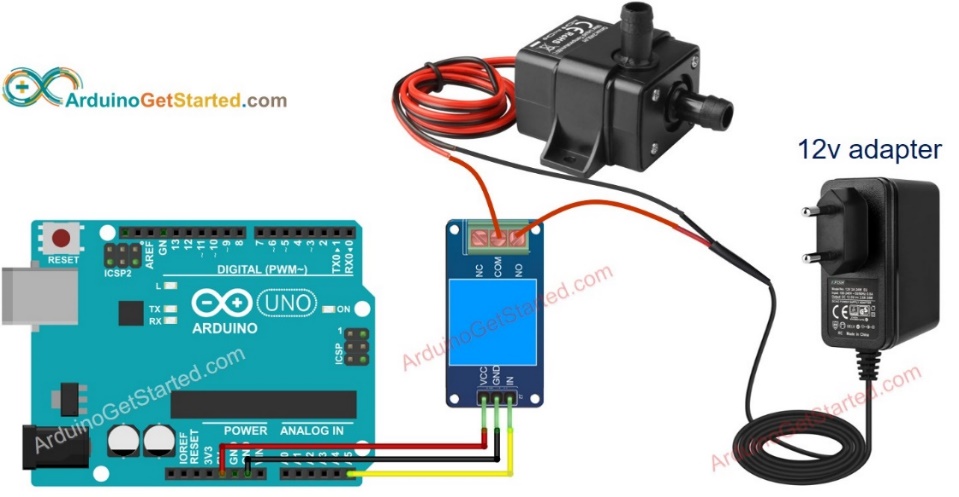
This Bluetooth module used is model HC-05, it works with serial communication as it transmits and receives data. The Bluetooth Module is used to send serial messages to the terminal on a mobile phone with notifications to inform the user of various messages like: if the Tank is Empty, if the tank is almost empty, if it’s raining so the plant doesn’t need to be watered.

## Rain Drop Sensor

The rain sensor detects water that comes short circuiting the tape of the printed circuits. The sensor acts as a variable resistance that will change status: the resistance increases when the sensor is wet and the resistance is lower when the sensor is dry. In this circuit, if the rain sensor detects rain, it notifies the user that plant is already watered from rain, and it will not activate the water pump to water the plant.

## Water Pump and Relay

The water pump requires 12V to get activated and requires a programmable switch to open/close it when needed. So, it is connected to a replay and a 12 V adaptor that takes from a direct power source then the relay waits for its command at a certain moisture level to get activated for certain amount of time to open the water pump and water the plant.



# Difficulties Faced

The only difficulty faced was in the simulation:

First, the water pump motor was unavailable so it was replaced with a servo motor only for the schematic shape

Second, the simulator used (proteus) does not allow the components to connect to the voltage pin or the GND pin and it does not even have a breadboard, so it was replaced by making a 5v battery as a common point to connect all components on and a ground as a common point to connect all ground pins of the components.

# References

|  |  |
| --- | --- |
| [1] | L. M. Engineers, “How Soil Moisture Sensor Works,” Last Minute Engineers, [Online]. Available: https://lastminuteengineers.com/soil-moisture-sensor-arduino-tutorial/. [Accessed 1 June 2021]. |
| [2] | Polygon, “How Humidity Affects the Growth of Plants,” Polygon, [Online]. Available: https://www.polygongroup.com/en-US/blog/how-humidity-affects-the-growth-of-plants/#:~:text=As%20plants%20transpire%2C%20the%20humidity,draw%20nutrients%20from%20the%20soil.. [Accessed 1 June 2021]. |