Smart water Management

About the project

Water is a **valuable resource**, and water shortages are a serious problem in many parts of the world. The problem can be made worse by people who waste water; for example, by watering a garden or using sprinklers on their lawn**. How can you help conserve water and prevent such waste?** One way is to build an electronic soil moisture sensor. This project will show you how to build a circuit that indicates whether soil is wet or dry, but the circuit itself is unprotected. It will be given by engineering solution, like **a waterproof** carrying case that turns the basic circuit into a useful, portable soil moisture sensor.

We interfaced the ESP32 with **a moisture sensor, temperature sensor, air humidity sensor, water flow sensor, and solenoid valve**. Using the data from these sensors, the **ESP32** determines when to open the solenoid valve. The solenoid valve controls the flow of water into the pipes of the drip irrigation system.

The efficient Water management practices based on the monitoring of the moisture in the soil provide a great benefit for the appropriate amount of water applied in the fields. This is a design and development of a soil moisture sensor and a response monitoring system. The probes used in this sensor are made of **nickel** which is an **anti-corrosive** and **robust material** for use in agricultural and garden related applications. The response monitoring system measure the moisture of the soil, compare it with the desired values given by the user and **generate alert** if soil moisture goes below desired value. It helps in problems related to growing of crops in which irrigation is required at irregular interval. It is also helpful in monitoring of soil moisture in golf fields.

Components used:

*DHT11 sensor

*Soil Moisture sensor

*Gsm Modem

*ultrasonic sensor

*Humidity and temperature sensor

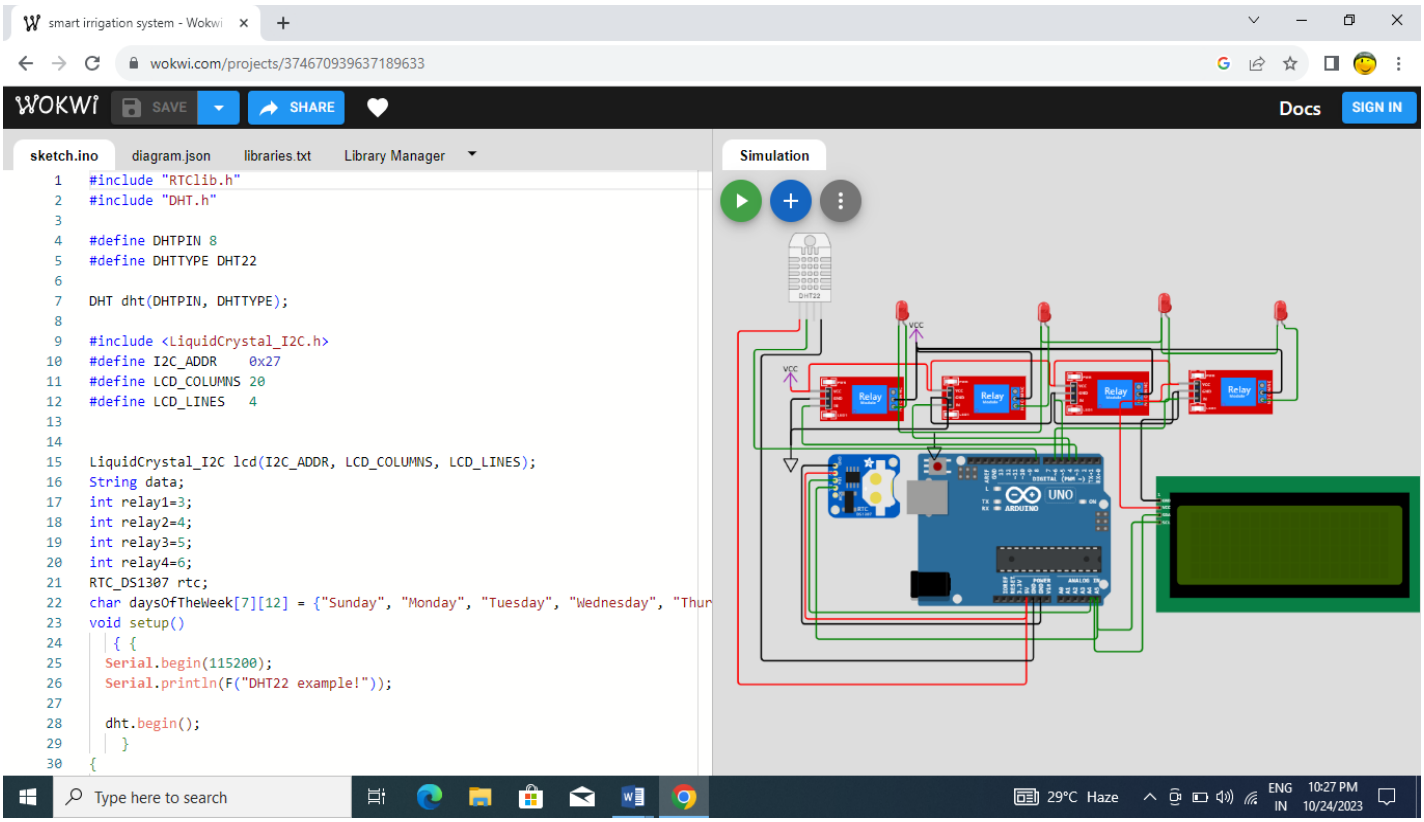
*Bluetooth Module

*Adaptor

*Connecting Jumper

*Peristalic pump

Connected components:



Aurdinon Code:

```
include "RTClib.h"
#include "DHT.h"

#define DHTPIN 8
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

#include <LiquidCrystal_I2C.h>
#define I2C_ADDR    0x27
#define LCD_COLUMNS 20
#define LCD_LINES   4


LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);
String data;
int relay1=3;
int relay2=4;
int relay3=5;
int relay4=6;
RTC_DS1307 rtc;
char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday",
"Thursday", "Friday", "Saturday"};
void setup()
   { {
  Serial.begin(115200);
  Serial.println(F("DHT22 example!"));

  dht.begin();
    }
{
  Serial.begin(115200);
   lcd.init();
  lcd.backlight();
  lcd.setCursor(3,0);
  lcd.print("welcome to");
  lcd.setCursor(2,1);
  lcd.print("SMART FARMING");
  delay(4000);
  pinMode(relay1, OUTPUT);
```

```
  pinMode(relay2, OUTPUT);
  pinMode(relay3, OUTPUT);
  pinMode(relay4, OUTPUT);
  Serial.println("welcome to my project");
  delay(500);
  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    Serial.flush();
    abort();
  }
  lcd.clear();

}
   }


void loop () {
   {
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  // Check if any reads failed and exit early (to try again).
  if (isnan(temperature) || isnan(humidity)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  Serial.print(F("Humidity: "));
  Serial.print(humidity);
  Serial.print(F("%  Temperature: "));
  Serial.print(temperature);
  Serial.println(F("°C "));
  lcd.setCursor(0,3);
    lcd.print("temp:");
    lcd.println(temperature);
    lcd.setCursor(10,3);
    lcd.print("hum:");
    lcd.println(humidity);
  delay(2000);
}



  DateTime now = rtc.now();
```

```
Serial.print("Current time: ");
Serial.print(now.year(), DEC);
Serial.print('/');
Serial.print(now.month(), DEC);
Serial.print('/');
Serial.print(now.day(), DEC);
Serial.print(" (");
Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
Serial.print(") ");
Serial.print(now.hour(), DEC);
Serial.print(':');
Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.println();
Serial.println();
delay(3000);
lcd.setCursor(3,0);
lcd.print("Time:");
lcd.print(now.hour(), DEC);
lcd.print(':');
lcd.print(now.minute(), DEC);
lcd.print(':');
lcd.print(now.second(), DEC);


if((now.second()> 1) && (now.second()<15))
{
lcd.setCursor(0,1);
lcd.print("Relay1:ON ");

 Serial.println("relay1 is on");
 digitalWrite(relay1, HIGH);
}
else{
  lcd.setCursor(0,1);
  lcd.print("Relay1:Off");
  digitalWrite(relay1,LOW);
}

 if((now.second()> 20) && (now.second()<30))
{
lcd.setCursor(10,1);
lcd.print("Relay2:ON ");
 Serial.println("relay2 is on");
```

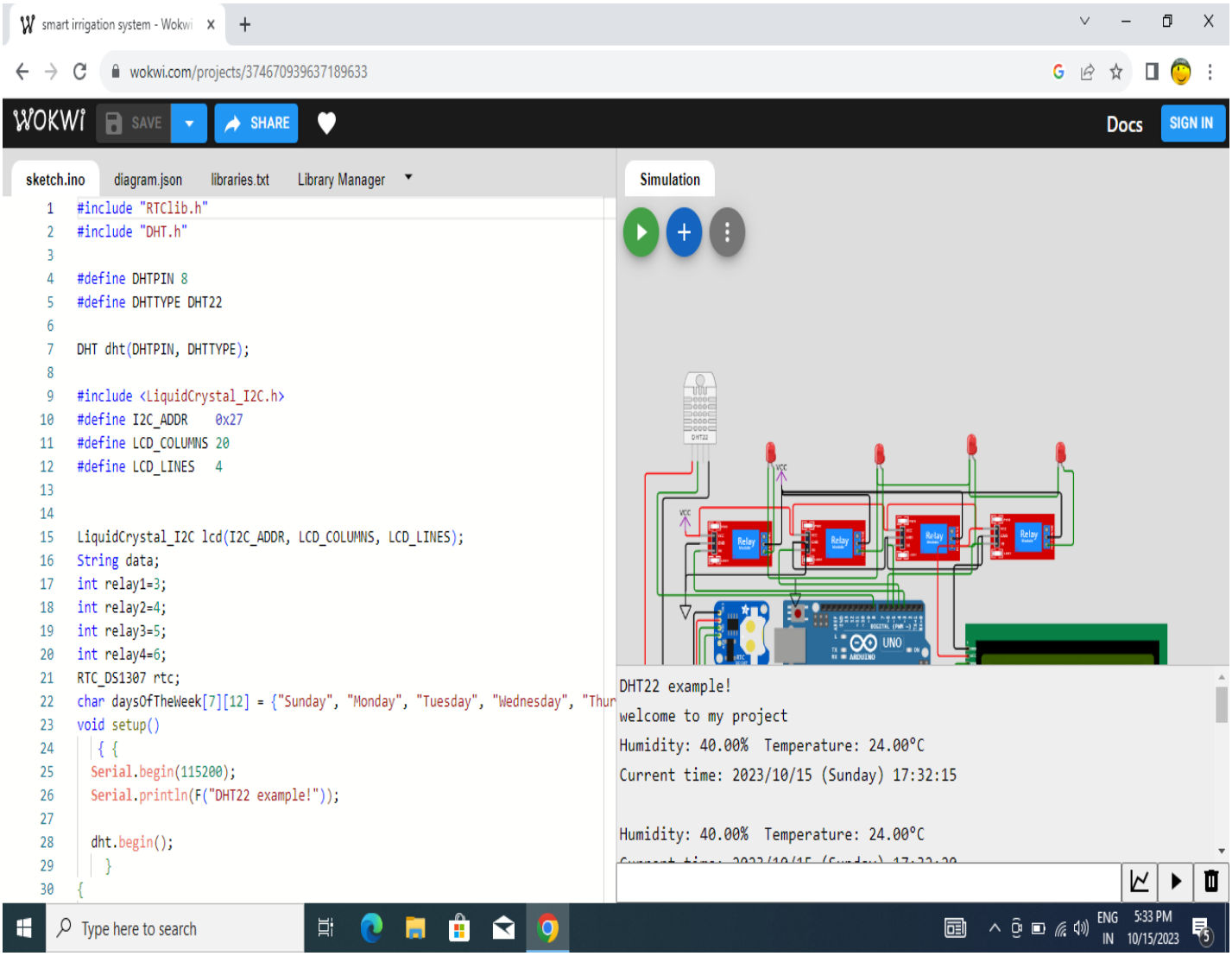
```
  else{
   lcd.setCursor(10,1);
  lcd.print("Relay2:OFF");
  digitalWrite(relay2,LOW);
  }
  if((now.second()> 35) && (now.second()<45))
  {
   lcd.setCursor(0,2);
  lcd.print("Relay3:ON ");
   Serial.println("relay3 is on");
   digitalWrite(relay3, HIGH);
  }
  else{
    lcd.setCursor(0,2);
  lcd.print("Relay3:OFF");
  digitalWrite(relay3,LOW);
  }
   if((now.second()> 50) && (now.second()<59))
  {
    lcd.setCursor(10,2);
  lcd.print("Relay4:ON ");
   Serial.println("relay4 is on");
   digitalWrite(relay4, HIGH);
  }
  else{
    lcd.setCursor(10,2);
  lcd.print("Relay4:OFF");
  digitalWrite(relay4,LOW);
  }
}
```

Python code:

```
from quarky import *

import iothouse

house = iothouse.iothouse()

import time

def Display_Soil_Moisture_Level():

 quarky.cleardisplay()

 Moisture = house.readmoisture()
```

```
  for row in range(1, 6):
    for column in range(1, 8):
      if ((((row - 1) * 20) + (column * (10 / 7))) < Moisture):
        quarky.setled(column, row, [0, 255, 0], 33)
def Watering_Animation():
  quarky.drawpattern("jfjjjfjjdjjjjjfdfjjjjjfjjjjjjjjjjj")
  time.sleep(0.2)
  quarky.drawpattern("jjjjfdfjfjjjfjjdjjjjjfdfjjjjjfjjjjj")
  time.sleep(0.2)
  quarky.drawpattern("jjjjjdjjjjjfdfjfjjjfjjdjjjjjfdfjjjj")
  time.sleep(0.2)
    quarky.drawpattern("jjjjjfjjjjjjdjjjjjfdfjfjjjfjjdjjjjj")
  time.sleep(0.2)
  quarky.drawpattern("jjjjjjjjjjjjfjjjjjjdjjjjjfdfjfjjjfj")
  time.sleep(0.2)
    quarky.drawpattern("jfjjjjjjjjjjjjjjjjfjjjjjjdjjjjjfdf")
  time.sleep(0.2)
  quarky.drawpattern("fdfjjjjjfjjjjjjjjjjjjjjjfjjjjjjdj")
  time.sleep(0.2)
  quarky.drawpattern("jdjjjjjfdfjjjjjfjjjjjjjjjjjjjjjfj")
  time.sleep(0.2)
while True:
  Display_Soil_Moisture_Level()
house.setmoisturepin("A2")
```

output:


smart irrigation system - Wokwi
wokwi.com/projects/374670939637189633
WOKWI
SAVE
SHARE
Docs
SIGN IN
sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
#include "RTClib.h"
#include "DHT.h"
#define DHTPIN 8
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
#include <LiquidCrystal_I2C.h>
#define I2C_ADDR 0x27
#define LCD_COLUMNS 20
#define LCD_LINES 4
LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);
String data;
int relay1=3;
int relay2=4;
int relay3=5;
int relay4=6;
RTC_DS1307 rtc;
char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thur
void setup()
{ {
Serial.begin(115200);
Serial.println(F("DHT22 example!"));
dht.begin();
}
{
DHT22 example!
welcome to my project
Humidity: 40.00% Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:15
Humidity: 40.00% Temperature: 24.00°C
Type here to search
ENG
IN
5:33 PM
10/15/2023

smart irrigation system - Wokwi

wokwi.com/projects/374670939637189633

WOKWI SAVE SHARE Docs SIGN IN

sketch.ino | diagram.json | libraries.txt | Library Manager

```
#include "RTClib.h"
#include "DHT.h"

#define DHTPIN 8
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

#include <LiquidCrystal_I2C.h>
#define I2C_ADDR    0x27
#define LCD_COLUMNS 20
#define LCD_LINES   4


LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);
String data;
int relay1=3;
int relay2=4;
int relay3=5;
int relay4=6;
RTC_DS1307 rtc;
char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thur
void setup()
  { {
  Serial.begin(115200);
  Serial.println(F("DHT22 example!"));

  dht.begin();
  }
{
```

Simulation

00:41.225 101%

```
Humidity: 40.00%  Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:25

relay2 is on
Humidity: 40.00%  Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:30

Humidity: 40.00%  Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:35

Humidity: 40.00%  Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:40

relay3 is on
Humidity: 40.00%  Temperature: 24.00°C
Current time: 2023/10/15 (Sunday) 17:32:45
```

Type here to search

ENG IN 5:32 PM 10/15/2023