**20CSL62 INTERNET OF THINGS AND CLOUD LABORATORY**

**SMART IRRIGATION SYSTEM**

**MINI PROJECT REPORT**

**Submitted by**

**RENESH RAJA S M**

**20CSR169**

**NAVANEETH S B**

**20CSR134**

**PRAVEEN RAJ KUMAR T**

**20CSL260**

***in partial fulfillment of the requirements for the award of the degree***

***of***

# BACHELOR OF ENGINEERING IN

**COMPUTER SCIENCE AND ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**KONGU ENGINEERING COLLEGE**

**(Autonomous)**

**PERUNDURAI ERODE – 638 060**

# ABSTRACT

In the 21st century, the AI industry has seen huge development. Nowadays, in farming, land residential areas are located in different places for people working in various jobs. So, maintaining the farming land is so difficult for them. Even for people who have both areas located in the same area, find it difficult to maintain. Our project helps to keep in check of the farming land whenever they want and can irrigate their land whenever they want, and irrespective of the place they are. They can make it possible through a connected internet. As this device will help to check the moisture content on the farming field, and if the water content is low, it will automatically sense and inlet the water out. If the water content reaches a certain limit, it stops the motor, and the process will be completed.

# COMPONENTS REQUIRED

* **Ardiuno esp8266**– The Arduino ESP8266 is a development board that can be used for creating Internet of Things (IoT) projects. It is based on the ESP8266 Wi-Fi module, which enables it to connect to the internet and communicate with other devices.
* **Moisture Sensor**– A moisture sensor is an electronic device that measures the moisture content of a material. It is used in a variety of applications, including agriculture, construction, and environmental monitoring.
* **Relay 12V** – A relay is an electrical component that allows a low-power signal to control a high-power circuit. A 12V relay is designed to switch a 12V DC circuit on or off using a small control signal.

* **Connecting Wires**– to connect two points in an electrical circuit.
* **Light Emitting Diode (LED)** – semiconductor device, which can emit light when an electrical current passes through it.
* **Breadboard** – to form simple electrical connections among different components.
* **Adapter 12v dc** - Adapter 12V DC power supply is an adapter designed to supply precisely 12 Volts of direct current to a device. The voltage supplied must precisely match the requirements of the equipment.

# HARDWARE SETUP

# 

**CODING**

#include <ESP8266WiFi.h>

#include <WiFiClientSecure.h>

#include <UniversalTelegramBot.h>

String chat\_id = " 1357935908";

// Enter your WiFi credentials

const char\* ssid = "Renish";

const char\* password = "12345678";

// Enter your Telegram Bot token (given by BotFather)

const char\* telegramToken = "6223321214:AAF9IKUekjAU0wUMAlnK99yB\_j-kQPy3rEQ";

// Initialize Telegram bot

WiFiClientSecure client;

UniversalTelegramBot bot(telegramToken, client);

const int ledPin = D5;

const int motorPin = D6;

const int moistureSensorPin = A0;

void setup() {

// Connect to WiFi

Serial.begin(115200);

WiFi.mode(WIFI\_STA);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

Serial.println("Connecting to WiFi...");

delay(1000);

}

Serial.println("WiFi connected!");

// Set up SSL connection for Telegram bot

client.setInsecure();

pinMode(ledPin, OUTPUT);

pinMode(motorPin, OUTPUT);

pinMode(moistureSensorPin, INPUT);

}

void loop() {

// Check for incoming messages

int numNewMessages = bot.getUpdates(bot.last\_message\_received + 1);

while (numNewMessages) {

Serial.println("New message received.");

handleNewMessages(numNewMessages);

numNewMessages = bot.getUpdates(bot.last\_message\_received + 1);

}

// Check the moisture sensor data and turn on/off the motor if needed

int moisture = analogRead(moistureSensorPin);

if (moisture < 20) {

digitalWrite(motorPin, HIGH);

bot.sendMessage(chat\_id, "Motor is now on.");

} else {

digitalWrite(motorPin, LOW);

bot.sendMessage(chat\_id, "Motor is now off.");

}

delay(1000);

}

void handleNewMessages(int numNewMessages) {

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

String chat\_id = String(bot.messages[i].chat\_id);

String text = bot.messages[i].text;

// Turn on the LED if the message is "/on"

if (text == "/on") {

digitalWrite(ledPin, HIGH);

bot.sendMessage(chat\_id, "LED is now on.");

}

// Turn off the LED if the message is "/off"

else if (text == "/off") {

digitalWrite(ledPin, LOW);

bot.sendMessage(chat\_id, "LED is now off.");

}

// Turn on the motor if the message is "/motor\_on"

else if (text == "/motor\_on") {

digitalWrite(motorPin, HIGH);

bot.sendMessage(chat\_id, "Motor is now on.");

}

// Turn off the motor if the message is "/motor\_off"

else if (text == "/motor\_off") {

digitalWrite(motorPin, LOW);

bot.sendMessage(chat\_id, "Motor is now off.");

}

// Send an error message for any other message

else {

bot.sendMessage(chat\_id, "Invalid command. Please type /on, /off, /motor\_on, or /motor\_off.");

}

}

}

# OUTPUT

# 

**CONCLUSION**

Overall, this IOT-based Smart irrigation application would help eliminate the major problem for people in irrigating their agricultural farms. In the future, it can be supported with a drone automated system to provide more functions.