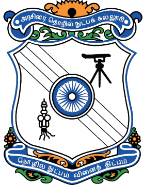


****

**GOVERNMENT COLLEGE OF TECHNOLOGY COIMBATORE-641013**

**DEPARTMENT OF ECE**

**SMART WATER FOUNTAINS**

**PHASE IV**

**TEAM MEMBERS**

1. SATHIYA PRAKASH B L -71772114138

2. KRISHNA PRAVEEN S -71772114302

3. MONISHA V - 71772114304

4. SUDHARANI K – 71772114L02

**SMART WATER FOUNTAINS**

**PHASE IV:STEP BY STEP PROCEDURE:**

Step-by-step procedure of how to simulate a Smart Water Fountain using the Wokwi simulator. we'll create a simple smart water fountain system that can be controlled remotely via a web interface.

**Components Needed:**

1. **NodeMCU ESP8266**

This will be our microcontroller.

1. **Water Pump**

To pump water from a container to the fountain.

1. **Relay Module**

To control the water pump.

1. **Ultrasonic Sensor (HC-SR04)**

To detect water level in the fountain.

1. **Wokwi Virtual Components**

These are virtual components you can add in Wokwi for the

web interface and simulation.

**Step 1: Create a Wokwi Account**

1. Go to the Wokwi website (https://wokwi.com/) and create a free account.

**Step 2: Set Up the Circuit in Wokwi Simulator**

1. Click on the "Create a New Project" button.

2. In the Wokwi Circuit Editor, add the NodeMCU ESP8266, Water Pump, Relay Module, and Ultrasonic Sensor to the canvas. Connect them appropriately with wires.

3. Add a "Button" and a "Range" element from the virtual components to the canvas. These will serve as the web interface controls for your water fountain.

**Simplified circuit layout in text:**

* **NodeMCU ESP8266:**

- Connect to Relay Module (Control Pin)

- Connect to Ultrasonic Sensor (Trigger and Echo Pins)

* **Relay Module:**

- Connect to Water Pump

* **Ultrasonic Sensor (HC-SR04):**

- VCC to 5V

- GND to GND

- Trig to NodeMCU GPIO (e.g., D2)

- Echo to NodeMCU GPIO (e.g., D3)

* **Button (Virtual Component):**

- Connect to NodeMCU GPIO (e.g., D4)

* **Range (Virtual Component):**

- Connect to NodeMCU GPIO (e.g., D5)

**Step 3: Write Arduino Code**

1. Write the Arduino code for your NodeMCU. we need to include libraries for the Ultrasonic Sensor and ESP8266 WiFi.

2. Set up a web server on your NodeMCU that listens for requests. When the button is pressed via the web interface, the server should activate the water pump.

3. Use the ultrasonic sensor to monitor the water level in the fountain and update the web interface accordingly.

**Arduino code:**

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <Ultrasonic.h>

const char\* ssid = "YourWiFiSSID";

const char\* password = "YourWiFiPassword";

const int trigPin = D2; // Trigger pin of Ultrasonic Sensor

const int echoPin = D3; // Echo pin of Ultrasonic Sensor

const int relayPin = D1; // Relay module control pin

Ultrasonic ultrasonic(trigPin, echoPin);

WiFiServer server(80);

void setup() {

pinMode(relayPin, OUTPUT);

digitalWrite(relayPin, LOW);

Serial.begin(115200);

WiFi.begin(ssid, password);

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

delay(1000);

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

}

server.begin();

}

void loop() {

WiFiClient client = server.available();

if (client) {

String request = client.readStringUntil('\r');

if (request.indexOf("/on") != -1) {

digitalWrite(relayPin, HIGH); // Turn the pump on

delay(2000); // Run the pump for 2 seconds

digitalWrite(relayPin, LOW); // Turn the pump off

}

client.flush();

}

// Check water level

float distance = ultrasonic.read();

if (distance < 10) {

// Water is low, update the web interface

// You can send an HTML response to the client here

}

}

**Step 4: Simulation**

1. Save your circuit and code in Wokwi.

2. Click the "Simulate" button to start the simulation.

3. You can interact with the virtual components in the simulation by clicking on the web interface controls (Button and Range).

With this setup, we can simulate a Smart Water Fountain that can be remotely controlled and monitors the water level in the fountain. This is a simplified example, and we can expand and customize it as needed for your project.