
SMART WATER FOUNTAIN – PHASE III

INTRODUCTION

In this project, we have created a Smart Water Fountain system that can be controlled remotely via a web interface. The system includes components such as NodeMCU ESP8266, Water Pump, Relay Module, and Ultrasonic Sensor (HC-SR04). The simulation was done using the Wokwi simulator, allowing us to test the functionality of the system virtually.

COMPONENTS USED

1. NodeMCU ESP8266: Microcontroller board.
2. Water Pump: Used to pump water from a container to the fountain.
3. Relay Module: Controls the water pump.
4. Ultrasonic Sensor (HC-SR04): Detects water level in the fountain.
5. Wokwi Virtual Components: Virtual elements used for creating the web interface and simulation.

SIMULATION SETUP

1. Wokwi Account: Created a Wokwi account to access the simulation platform.
2. Circuit Configuration: Set up the circuit in the Wokwi Circuit Editor, connecting NodeMCU ESP8266, Water Pump, Relay Module, and Ultrasonic Sensor. Virtual components like Button and Range were added for the web interface.

CIRCUIT CONNECTIONS (SIMPLIFIED)

1. NodeMCU ESP8266:
 - Connected to Relay Module (Control Pin)
 - Connected to Ultrasonic Sensor (Trigger and Echo Pins)

2. Relay Module:

- Controls the Water Pump

3. Ultrasonic Sensor (HC-SR04):

- VCC to 5V
- GND to GND
- Trig to NodeMCU GPIO (e.g., D2)
- Echo to NodeMCU GPIO (e.g., D3)

4. Button (Virtual Component):

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

5. Range (Virtual Component):

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

ARDUINO CODE

The Arduino code was developed to control the water pump based on user input from the web interface and monitor the water level using the Ultrasonic Sensor.

```
#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
}
}

```

SIMULATION AND TESTING

1. Code Implementation: Implemented the Arduino code to control the water pump and monitor water level.
2. Simulation: Clicked the "Simulate" button in Wokwi to start the simulation.
3. Testing: Interacted with the virtual components in the simulation to test the system's functionality.

CONCLUSION

The Smart Water Fountain system simulation was successfully carried out using the Wokwi simulator. The system demonstrated remote control capabilities and real-time monitoring of the water level. This simulation provides a basis for further development and implementation in real-world IoT projects.
