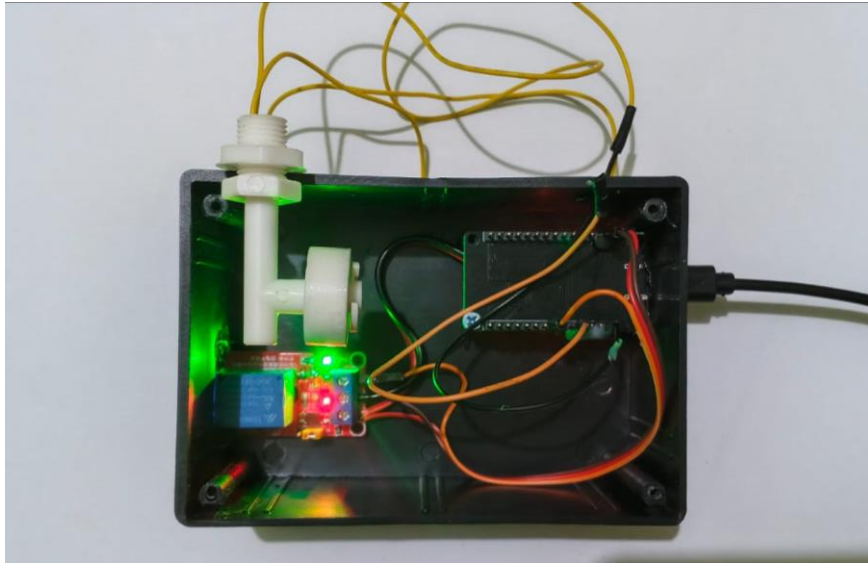


LAMPIRAN

Dokumentasi Penelitian





Sketch Lengkap

```

#include <WiFi.h>
#include <PubSubClient.h>

// ===== WIFI =====
const char* ssid = "smartwater";
const char* password = "12345678";

// ===== MQTT =====
const char* mqtt_server = "broker.hivemq.com";

const char* topicCommand = "water/tian/pump/command";
const char* topicStatus = "water/tian/pump/status";
const char* topicLevel = "water/tian/tank/level";

WiFiClient espClient;
PubSubClient client(espClient);

// ===== PIN =====
#define RELAY_PIN 5
#define FLOAT_PIN 4

bool lastFloatState = HIGH;
bool pumpState = false;

// =====
// CONNECT WIFI
// =====
void setup_wifi() {
  delay(10);
  Serial.println("Connecting WiFi...");

```

```

WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}

Serial.println("\nWiFi Connected");
}

// =====
// MQTT CALLBACK
// =====
void callback(char* topic, byte* payload, unsigned int
length) {
  String message;

  for (int i = 0; i < length; i++) {
    message += (char)payload[i];
  }

  Serial.print("Message arrived: ");
  Serial.println(message);

  if (String(topic) == topicCommand) {
event
    if (message == "ON") {
      digitalWrite(RELAY_PIN, LOW); // aktif LOW relay
      pumpState = true;
      client.publish(topicStatus, "ON");
    }

    if (message == "OFF") {
      digitalWrite(RELAY_PIN, HIGH);
      pumpState = false;
      client.publish(topicStatus, "OFF");
    }
  }
}

// =====
// RECONNECT MQTT
// =====
void reconnect() {
  while (!client.connected()) {
    Serial.print("Connecting MQTT...");
    if (client.connect("ESP32-TIAN")) {
      Serial.println("connected");
      client.subscribe(topicCommand);
    }
  }
}

```

```

    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      delay(2000);
    }
  }
}

// =====
// SETUP
// =====
void setup() {
  Serial.begin(115200);

  pinMode(RELAY_PIN, OUTPUT);
  pinMode(FLOAT_PIN, INPUT_PULLUP);

  digitalWrite(RELAY_PIN, HIGH); // default OFF

  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}

// =====
// LOOP
// =====
void loop() {

  if (!client.connected()) {
    reconnect();
  }

  client.loop();

  // ===== FLOAT SWITCH =====
  bool floatState = digitalRead(FLOAT_PIN);

  if (floatState != lastFloatState) {
    delay(200); // debounce

    if (floatState == LOW) {
      Serial.println("Air FULL");
      client.publish(topicLevel, "FULL");
    } else {
      Serial.println("Air LOW");
      client.publish(topicLevel, "LOW");
    }
  }
}

```

```
        lastFloatState = floatState;
    }
}
```