

LAPORAN UAS

Sistem Kontrol Pompa Air Berbasis ESP32, Web Server, dan MQTT

Diajukan untuk memenuhi ujian akhir semester mata kuliah *Sistem Mikroprosesor*

Dosen Pengampu :

Muhammad Ikhwan Fathulloh, S.Kom.



Disusun oleh :

Aldi Rizkiansyah 23552011130

**FAKULTAS TEKNIK INDUSTRI KREATIF
DEPARTEMEN TEKNIK INFORMATIKA
UNIVERSITAS TEKNOLOGI BANDUNG**

2026

1. Pendahuluan

Perkembangan teknologi Internet of Things (IoT) memungkinkan berbagai perangkat untuk saling terhubung melalui jaringan internet. Salah satu penerapannya adalah sistem kontrol perangkat elektronik secara jarak jauh. Pada proyek ini, dibuat sebuah sistem kontrol pompa air berbasis ESP32 yang dapat dikendalikan melalui website lokal dan layanan MQTT.

Sistem ini memungkinkan pengguna untuk:

- Mengontrol pompa air melalui web browser
- Mengontrol pompa menggunakan tombol fisik
- Mengirim dan menerima data melalui MQTT
- Mengatur koneksi WiFi secara dinamis

2. Tujuan

Tujuan pembuatan sistem ini adalah:

1. Membuat sistem kontrol pompa air berbasis ESP32.
2. Mengimplementasikan Web Server sebagai antarmuka pengguna.
3. Menggunakan MQTT sebagai media komunikasi data.
4. Menyimpan konfigurasi WiFi menggunakan memori internal.
5. Mengintegrasikan tombol fisik sebagai kontrol manual.

3. Perangkat dan Software

3.1 Perangkat Keras

- ESP32
- Relay / Pompa Air
- Tombol Push Button
- Kabel jumper
- Sumber daya

3.2 Perangkat Lunak

- Arduino IDE
- Library WiFi.h
- Library WebServer.h
- Library Preferences.h
- Library ArduinoJson.h
- Library PubSubClient.h

4. Konsep Kerja Sistem

Sistem bekerja dengan cara:

1. ESP32 menyala dan membaca data WiFi dari memori.
2. Jika belum ada data WiFi, ESP32 masuk ke mode Access Point.
3. Pengguna mengatur WiFi melalui web.
4. ESP32 terhubung ke internet.
5. ESP32 terhubung ke server MQTT.
6. Pengguna dapat mengontrol pompa melalui web, MQTT, atau tombol.
7. Status pompa dikirim ke server MQTT.

5. Penjelasan Program

5.1 Library

```
#include <WiFi.h>
#include <WebServer.h>
#include <Preferences.h>
#include <ArduinoJson.h>
#include <PubSubClient.h>
```

Library digunakan untuk:

- WiFi.h: Mengatur koneksi WiFi
- WebServer.h: Membuat web server
- Preferences.h: Menyimpan data permanen
- ArduinoJson.h: Mengolah data JSON
- PubSubClient.h: Komunikasi MQTT

5.2 Konfigurasi Pin

```
#define POMPA_PIN 2
#define BTN_PIN 0
• POMPA_PIN: Pin untuk mengontrol pompa
• BTN_PIN: Pin untuk tombol manual
```

5.3 Konfigurasi MQTT

```
const char* mqttServer = "public.cloud.shiftr.io";
```

Digunakan untuk menentukan server, port, username, dan password MQTT.

Topic:

- iot/esp32/pompa/control
- iot/esp32/pompa/status

5.4 Variabel Global

```
String ssid, password;  
bool pompaState = false;
```

Digunakan untuk menyimpan data WiFi dan status pompa.

5.5 Interrupt Tombol

```
void IRAM_ATTR togglePompa()
```

Fungsi ini akan berjalan saat tombol ditekan. Fungsi ini mengubah status pompa dan mengirim data ke MQTT.

5.6 Callback MQTT

```
void mqttCallback(...)
```

Fungsi ini menerima pesan dari MQTT. Jika menerima perintah ON/OFF, maka pompa akan dikontrol sesuai perintah.

5.7 Koneksi MQTT

```
void connectMQTT()
```

Fungsi ini menghubungkan ESP32 ke server MQTT dan melakukan subscribe pada topic kontrol.

5.8 Task MQTT

```
void taskMQTT(void *pv)
```

Task ini dijalankan menggunakan FreeRTOS agar MQTT berjalan secara paralel dengan program utama.

5.9 Web Server

Halaman web disimpan menggunakan PROGMEM agar menghemat memori.

Fitur web:

- Tombol ON/OFF pompa
- Status WiFi

- Alamat IP
- Pengaturan WiFi

5.10 Setup()

Fungsi setup digunakan untuk:

- Inisialisasi Serial
- Pengaturan pin
- Membaca data WiFi
- Menjalankan server
- Menjalankan task MQTT

5.11 Routing Web

Beberapa endpoint:

- / : Halaman utama
- /pompa/on : Menyalakan pompa
- /pompa/off : Mematikan pompa
- /wifi : Menyimpan WiFi
- /status : Mengirim status sistem

5.12 Loop()

```
void loop() {  
    server.handleClient();  
}
```

Loop digunakan untuk melayani permintaan dari client web.

5.13 Koneksi WiFi

```
void connectWiFi()
```

Fungsi ini menghubungkan ESP32 ke jaringan WiFi.

Jika gagal, sistem masuk ke mode AP.

5.14 Mode Access Point

```
void startAPMode()
```

Mode ini digunakan saat ESP32 belum memiliki data WiFi.

SSID: ESP32-Config

6. Pengujian Sistem

Pengujian dilakukan dengan:

1. Mengakses web server melalui browser.
2. Menekan tombol ON/OFF.
3. Mengirim perintah melalui MQTT.
4. Mematikan WiFi untuk menguji mode AP.

Hasil:

- Sistem dapat berjalan dengan baik.
- Pompa dapat dikontrol dari berbagai media.
- Status dapat dipantau secara real-time.
- Bergantung pada koneksi internet

7. Kesimpulan

Berdasarkan hasil perancangan dan pengujian, sistem kontrol pompa berbasis ESP32 ini dapat berjalan dengan baik. Sistem mampu mengontrol pompa melalui web, MQTT, dan tombol fisik. Selain itu, sistem juga dapat menyimpan konfigurasi WiFi secara permanen.

Proyek ini dapat dikembangkan lebih lanjut dengan menambahkan fitur keamanan, sensor, dan aplikasi mobile.