

Laporan Tugas

Kontrol Motor DC Berbasis ESP32 dengan MQTT, PID, dan Monitoring RPM



Oleh

23081010297

Adiel Fatich Ar Rafi

PROGRAM STUDI INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA TIMUR

SURABAYA

2025

BAB I

PENDAHULUAN

Motor DC merupakan komponen penting dalam sistem kendali dan otomasi. Pengendalian kecepatan motor DC dapat dilakukan dengan memanfaatkan mikrokontroler ESP32 yang memiliki kemampuan komunikasi WiFi. Dengan integrasi protokol MQTT, sistem ini dapat dikendalikan secara jarak jauh.

Selain itu, penerapan algoritma PID (Proportional-Integral-Derivative) digunakan untuk menjaga kestabilan kecepatan motor sesuai target RPM. Monitoring RPM dilakukan dengan sensor encoder sehingga kecepatan motor dapat diamati secara real-time.

Tujuan dari tugas ini adalah merancang dan mengimplementasikan sistem kontrol motor DC berbasis ESP32 dengan fitur kontrol jarak jauh melalui MQTT, penerapan PID untuk kestabilan kecepatan, serta monitoring RPM secara real-time.

BAB II

LANDASAN TEORI

2.1 Motor DC

Motor DC adalah aktuator listrik yang mengubah energi listrik menjadi energi mekanik berupa putaran. Kecepatan motor DC dapat dikendalikan dengan mengatur tegangan atau duty cycle PWM.

2.2 ESP32

ESP32 adalah mikrokontroler dengan fitur WiFi dan Bluetooth terintegrasi. ESP32 mendukung komunikasi MQTT sehingga dapat digunakan dalam sistem IoT untuk kontrol jarak jauh.

2.3 MQTT

MQTT (Message Queuing Telemetry Transport) adalah protokol komunikasi ringan berbasis publish-subscribe. MQTT digunakan untuk mengirim perintah kontrol motor dan menerima data RPM.

2.4 PID Controller

PID adalah algoritma kontrol yang terdiri dari tiga komponen:

- **Proportional (P):** merespons besar error.
- **Integral (I):** menghilangkan error jangka panjang.
- **Derivative (D):** meredam perubahan cepat.

BAB III

METODOLOGI

3.1 Perancangan Sistem

Sistem terdiri dari ESP32, motor DC, driver motor, sensor encoder, dan LED indikator. ESP32 menerima perintah melalui MQTT, menghitung RPM dari sensor encoder, kemudian mengatur duty cycle PWM menggunakan PID.

3.2 Implementasi Perangkat Lunak

- **Interrupt:** digunakan untuk menghitung pulsa encoder.
- **Fungsi calculateRPM():** menghitung RPM tiap interval waktu.

- **PID Control:** menyesuaikan duty cycle PWM agar RPM mendekati target.
- **MQTT Callback:** menerima perintah numeric setpoint atau perintah stop.
- **LED Indikator:** menunjukkan status WiFi dan motor.

3.3 Alur Kerja

1. ESP32 terhubung ke WiFi dan broker MQTT.
2. Perintah kontrol dikirim melalui MQTT.
3. ESP32 menghitung RPM dari sensor encoder.
4. PID controller menyesuaikan duty cycle PWM.
5. Data RPM dikirim kembali ke MQTT untuk monitoring.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Implementasi

- Sistem berhasil menerima perintah numeric RPM melalui MQTT.
- Kick-start duty cycle efektif mencegah motor berhenti saat transisi antar mode.
- RPM berhasil dihitung dan dikirim ke MQTT topic esp32/motor/speed.
- PID control menjaga RPM lebih stabil dibanding kontrol bang-bang.
- LED indikator berfungsi sesuai status sistem.

4.2 Pembahasan

Integrasi ESP32, MQTT, dan PID meningkatkan kestabilan kecepatan motor DC. Akurasi RPM sangat bergantung pada resolusi encoder. Encoder dengan jumlah pulsa lebih tinggi akan menghasilkan data RPM yang lebih halus dan akurat.

BAB V

KESIMPULAN

1. Sistem kontrol motor DC berbasis ESP32 dengan MQTT berhasil diimplementasikan.
2. RPM dapat dimonitor secara real-time menggunakan sensor encoder.
3. PID controller mampu menjaga kestabilan RPM sesuai setpoint.
4. Kick-start duty cycle efektif mencegah motor berhenti saat transisi antar mode.