

# Laporan Praktikum: Kontrol Motor DC dan Sensor Encoder Menggunakan ESP32

---

Nama : Wisanggeni Atthoriq Kuswirasatya  
NPM : 22081010127

---

## 1. Pendahuluan

Praktikum ini bertujuan untuk memahami prinsip kerja sistem kontrol motor DC menggunakan mikrokontroller ESP32. Sistem terdiri dari modul *development board* iMCLab yang mengintegrasikan ESP32, driver motor (H-Bridge), motor DC dengan *gearbox*, dan sensor kecepatan (Optical Encoder). Fokus utama eksperimen adalah identifikasi pin (*pinout mapping*), pengujian aktuator (motor), dan pembacaan sensor.

## 2. Eksperimen 1: Identifikasi Hardware dan Troubleshooting Pinout

### Deskripsi Eksperimen

Langkah pertama adalah menentukan pin GPIO ESP32 yang terhubung ke Driver Motor dan Sensor Encoder, mengingat dokumentasi skematik modul tidak tersedia secara spesifik. Dilakukan metode *scanning* sinyal digital menggunakan kode diagnostik untuk memicu semua kemungkinan pin output dan membaca pin input.

### Foto Hasil Eksperimen



### Analisis

Dari hasil pengujian diagnostik, ditemukan konfigurasi pin sebagai berikut:

#### 1. Motor Driver (L293D/L298N):

- **Pin Arah (Direction):** GPIO 26 dan GPIO 27. Mengubah logika HIGH/LOW pada pin ini membalik arah putaran motor.
- **Pin Enable (PWM/Speed):** GPIO 12 dan GPIO 14. Pin ini berfungsi sebagai "pedal gas". Tanpa memberikan sinyal HIGH pada pin ini, motor tidak akan menerima daya meskipun pin arah sudah diset. Awalnya motor gagal berputar karena pin Enable ini belum diidentifikasi.

## 2. Sensor Encoder:

- Terhubung pada salah satu pin input (GPIO 34/35/36/39).

## 3. Eksperimen 2: Pengujian Aktuator (Motor Run)

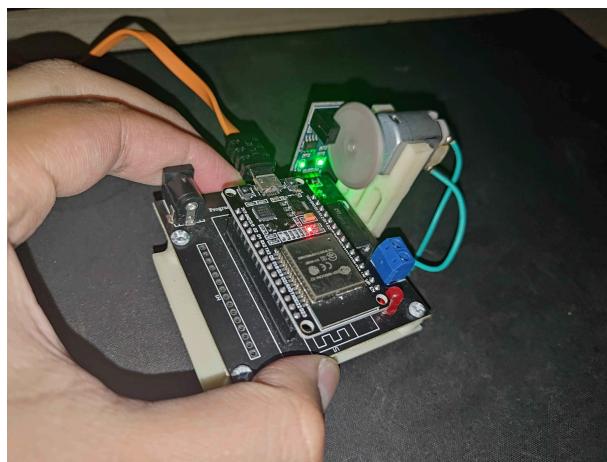
### Deskripsi Eksperimen

Setelah pinout diidentifikasi, dilakukan pengujian untuk menjalankan motor secara kontinu (Continuous Run). Kode program Run.ino diunggah untuk memberikan logika HIGH pada pin Enable dan konfigurasi arah maju pada pin Motor.

### Kode Program (Snippet)

```
// Konfigurasi Pin Motor
digitalWrite(MOTOR_PIN_A, HIGH);
digitalWrite(MOTOR_PIN_B, LOW);
// Mengaktifkan Enable (Penting)
digitalWrite(ENABLE_PIN_1, HIGH);
digitalWrite(ENABLE_PIN_2, HIGH);
```

### Foto Hasil Eksperimen



### Analisis

Motor berhasil berputar secara konsisten. Hal ini membuktikan bahwa:

1. Rangkaian driver motor berfungsi normal.
2. Catu daya (Power Supply) dari USB dan jack DC eksternal mampu mensuplai arus yang cukup untuk memutar beban motor.
3. Logika kontrol arah bekerja sesuai prinsip H-Bridge: Arus mengalir dari Pin A ke B menyebabkan putaran searah jarum jam (atau sebaliknya).

## 4. Eksperimen 3: Pengujian Sensor Kecepatan (Optical Encoder)

### Deskripsi Eksperimen

Eksperimen ini bertujuan membaca umpan balik (*feedback*) kecepatan motor menggunakan sensor optik. Sensor

bekerja dengan memancarkan cahaya inframerah (IR) yang diputus-sambung oleh piringan berlubang (*slotted disc*) pada roda.

### Masalah dan Solusi

Awalnya, LED indikator sensor menyala solid (tidak berkedip) dan pembacaan di Serial Monitor tetap '0' meskipun roda berputar.

Analisis Masalah:

Penyebabnya adalah transparansi material roda encoder terhadap cahaya inframerah. Meskipun roda terlihat berwarna abu-abu/hitam oleh mata manusia, material plastiknya tembus pandang bagi sensor IR, sehingga sensor menganggap "tidak ada halangan".

Solusi:

Dilakukan "Cardboard Test" (uji manual dengan potongan kardus) yang berhasil memicu sensor. Solusi permanen adalah melapisi jeruji roda dengan material pekat cahaya (seperti selotip hitam atau spidol permanen) agar sensor dapat mendeteksi putaran (logika 1 dan 0 bergantian).

## 5. Eksperimen 4: Penghentian Sistem (Motor Stop)

### Deskripsi Eksperimen

Pengujian mekanisme *safety* atau penghentian sistem menggunakan kode Stop.ino.

### Analisis

Untuk menghentikan motor secara total via software, seluruh pin kontrol (Arah dan Enable) diberi logika LOW.

```
digitalWrite(26, LOW); // Arah A OFF
digitalWrite(27, LOW); // Arah B OFF
digitalWrite(12, LOW); // Enable 1 OFF
digitalWrite(14, LOW); // Enable 2 OFF
```

Dalam sistem kontrol tertutup (PID), fungsi ini penting untuk *emergency stop* atau saat *setpoint* kecepatan tercapai nol.

## 6. Kesimpulan

Melalui serangkaian eksperimen ini, sistem iMCLab berhasil dioperasikan:

1. Motor DC dapat dikendalikan (Start/Stop dan Arah) menggunakan GPIO 26, 27, dan 12.
2. Pentingnya memahami peran pin **Enable** pada driver motor; tanpa pin ini, logika arah saja tidak cukup untuk menggerakkan motor.
3. Permasalahan sensor optik seringkali bukan pada kerusakan elektronik, melainkan sifat fisik material (transparansi IR) atau kalibrasi sensitivitas potensiometer.