

BAB II

DASAR TEORI

2.1 Tinjauan Pustaka

Setelah melakukan studi literatur terhadap berbagai jurnal yang berkaitan dengan sistem gerbang otomatis, *Automatic Line Barrier* (ALB), sensor ultrasonik, mikrokontroler ESP32, serta metode *Finite State Machine* (FSM), diperoleh beberapa penelitian terdahulu yang menjadi dasar pengembangan penelitian ini. Kajian tersebut digunakan untuk mengidentifikasi kelebihan, keterbatasan, serta celah penelitian sehingga dapat menjadi landasan dalam merancang sistem kendali *Automatic Line Barrier* pada *prototype* gerbang tol *free flow*. Ringkasan penelitian terdahulu disajikan pada **Tabel 2.1**.

Tabel 2.1 Tinjauan pustaka

No	Judul Penelitian	Keterangan	Target Pembaruan
1	<i>Design and Development of an ESP32 Based Barrier Gate Control System</i> (JEAT, 2025)	Mengembangkan sistem <i>barrier gate</i> berbasis ESP32 sebagai pengendali utama untuk membuka dan menutup palang secara otomatis. Sistem telah mampu mengendalikan <i>barrier gate</i> , namun logika kendali masih bersifat prosedural dan belum menerapkan metode <i>Finite State Machine</i> (FSM).	Menerapkan metode <i>Finite State Machine</i> (FSM) sehingga proses deteksi kendaraan, verifikasi transaksi, pembukaan palang, hingga penutupan palang berlangsung secara deterministik dan lebih terstruktur.
2	Implementasi Sistem Pada	Mengimplementasikan <i>barrier gate</i> otomatis	Menambahkan sensor ultrasonik sebagai

No	Judul Penelitian	Keterangan	Target Pembaruan
	Automasi <i>Barrier Gate</i> Palang Pintu Parkir Menggunakan ESP32 dan RFID (Universitas Tanjungpura, 2025)	menggunakan ESP32 dan RFID sebagai media identifikasi kendaraan. Fokus penelitian berada pada sistem parkir sehingga belum membahas mekanisme kendali berbasis <i>state</i> maupun keamanan saat kendaraan melewati palang.	pendeteksi posisi kendaraan dan menerapkan FSM agar proses buka-tutup palang lebih aman dan sistematis.
3	Pengembangan Sistem Palang Pintu Otomatis Menggunakan RFID dan Sensor Ultrasonik (UNESA, 2025)	Mengintegrasikan RFID dan sensor ultrasonik pada sistem palang otomatis. Sensor ultrasonik digunakan sebagai pendeteksi kendaraan, namun belum menerapkan metode pengendalian yang mampu mengatur transisi kondisi sistem secara terstruktur.	Mengembangkan sistem dengan <i>Finite State Machine</i> (FSM) sehingga setiap kondisi operasi memiliki <i>state</i> dan transisi yang jelas untuk meningkatkan keandalan sistem.
4	<i>Prototype</i> Pengendalian Gerbang Geser Otomatis Berbasis Arduino Uno	Menggunakan sensor ultrasonik sebagai pendeteksi kendaraan dan motor servo sebagai aktuator gerbang otomatis.	Menggunakan ESP32 yang memiliki kemampuan komunikasi Wi-Fi serta menerapkan FSM sehingga sistem lebih

No	Judul Penelitian	Keterangan	Target Pembaruan
	(Syntax Literate, 2025)	Sistem masih menggunakan Arduino Uno dan logika kontrol sederhana sehingga fleksibilitas pengembangan masih terbatas.	responsif dan mudah dikembangkan menuju sistem IoT.
5	<i>Software Development of prototype Environmental Friendly RFID Based Automatic Toll Gate with Electronic Paperless Payment Notification</i> (Universitas Diponegoro)	Mengembangkan gerbang tol otomatis berbasis RFID implementasi <i>Electronic Toll Collection</i> (ETC). Penelitian berfokus pada identifikasi kendaraan dan sistem transaksi tanpa membahas mekanisme pengendalian <i>Automatic Line Barrier</i> .	Mengembangkan sistem kendali <i>Automatic Line Barrier</i> (ALB) berbasis sensor ultrasonik dan FSM sebagai mekanisme kontrol akses kendaraan pada <i>prototype gerbang tol free flow</i> .

2.2 Single Lane Free Flow (SLFF)

Gerbang tol (*toll gate*) adalah tempat pelayanan transaksi pembayaran tol antara pengguna jalan dan pengelola jalan tol. Sedangkan *Single Lane Free Flow* (SLFF) merupakan sistem transaksi tol berbasis teknologi yang memungkinkan kendaraan melintas di jalan tol tanpa harus berhenti di gerbang tol. Pada sistem ini, proses identifikasi kendaraan dan pembayaran tarif tol dilakukan secara otomatis menggunakan perangkat elektronik yaitu sensor. Dengan penerapan SLFF, kendaraan dapat tetap bergerak dengan kecepatan tertentu sehingga dapat mengurangi waktu antrean, meningkatkan kapasitas

jalan, serta meminimalkan kemacetan yang biasanya terjadi di gerbang tol akibat proses transaksi manual maupun elektronik konvensional. Oleh karena itu, SLFF dianggap sebagai solusi modern dalam meningkatkan efisiensi sistem operasional jalan tol di Indonesia. Dalam penerapannya bisa di lihat pada **Gambar 2.1**.



Gambar 2.1 Ilustrasi *Single Lane Free Flow* (SLFF) [7]

Pada **Gambar 2.1** merupakan ilustrasi dari sistem SLFF yang sudah diimplementasikan. Pada sistem transaksi manual, kendaraan diharuskan berhenti di gardu tol untuk melakukan pembayaran secara tunai melalui petugas. Proses ini meliputi penyerahan uang, perhitungan tarif, hingga pemberian kembalian, sehingga membutuhkan waktu relatif lebih lama dan berpotensi menimbulkan antrean kendaraan [8]. Seiring perkembangan teknologi, sistem pembayaran elektronik (*e-toll*) mulai diterapkan dengan memanfaatkan kartu berbasis RFID, di mana proses transaksi dilakukan secara otomatis melalui pembacaan kartu dan pemotongan saldo tanpa perhitungan manual [9]. Meskipun lebih cepat dan efisien dibandingkan sistem manual, metode ini masih mengharuskan kendaraan berhenti, sehingga belum sepenuhnya menghilangkan potensi kemacetan di gerbang tol [10].

2.3 Automatic Line Barrier (ALB)

Automatic Line Barrier (ALB) merupakan perangkat aktuator otomatis yang berfungsi untuk mengendalikan akses kendaraan melalui mekanisme pembukaan dan penutupan palang secara terkontrol. Dalam sistem gerbang tol

modern, ALB tidak hanya berperan sebagai penghalang fisik, tetapi juga menjadi bagian dari sistem kendali terintegrasi yang bekerja berdasarkan hasil deteksi kendaraan dan verifikasi transaksi. Berdasarkan uraian pada proposal, kendala yang sering terjadi pada sistem gerbang tol salah satunya disebabkan oleh gangguan pada ALB, seperti kegagalan membuka palang yang dapat memicu antrian kendaraan. Oleh karena itu, diperlukan sistem kendali yang mampu mengatur operasi ALB secara lebih stabil dan responsif.



Gambar 2.2 *Automatic Line Barrier (ALB)* [11]

Pada Gambar 2.2 merupakan model dari ALB yang sering digunakan pada gerbang tol. Pada penelitian ini, ALB diimplementasikan sebagai komponen *output* yang dikendalikan oleh mikrokontroler ESP32 melalui pendekatan *Finite State Machine* (FSM), sehingga setiap kondisi sistem seperti deteksi kendaraan, validasi transaksi, hingga kondisi aman untuk menutup palang dapat diatur secara terstruktur. Sensor ultrasonik digunakan sebagai input untuk mendeteksi keberadaan kendaraan, sedangkan sistem verifikasi berbasis IoT (*Firebase*) berfungsi sebagai pemicu keputusan pembukaan palang. Mekanisme ini sejalan dengan penelitian sebelumnya yang menyatakan bahwa sistem *barrier gate* bekerja berdasarkan sinyal dari unit kontrol setelah proses identifikasi atau validasi dilakukan [12]. Dengan integrasi tersebut, ALB tidak hanya berfungsi sebagai aktuator mekanis, tetapi juga sebagai bagian penting dalam sistem kendali yang menentukan keberhasilan operasional gerbang tol otomatis, serta menjadi tahap transisi menuju sistem *free flow* yang lebih efisien dan tanpa henti.

2.4 *Finite State Machine (FSM)*

Finite State Machine (FSM) merupakan salah satu model komputasi yang digunakan untuk merepresentasikan perilaku suatu sistem melalui sekumpulan keadaan (*state*) dan perpindahan antar keadaan (*transition*) yang dipicu oleh suatu masukan atau kejadian (*event*). Pada setiap saat sistem hanya berada pada satu *state* aktif, kemudian akan berpindah menuju *state* lainnya apabila memenuhi kondisi transisi yang telah ditentukan. Pendekatan ini banyak digunakan pada sistem *embedded* karena mampu menyederhanakan logika kendali yang kompleks menjadi alur kerja yang lebih sistematis dan mudah dipahami [6].

Menurut Chang dan Mobasser, FSM memodelkan perilaku sistem menggunakan representasi berbasis *state* sehingga proses perancangan, implementasi, dan pengujian perangkat lunak dapat dilakukan secara lebih terstruktur. Pendekatan ini sesuai jika diterapkan pada sistem yang memiliki urutan operasi dan kondisi kerja yang saling bergantung, seperti sistem kendali pada ALB. Pendekatan ini banyak digunakan pada sistem berbasis mikrokontroler karena mampu menyederhanakan logika kontrol yang kompleks menjadi lebih terorganisir dan mudah diimplementasikan [13].

Metode *Finite State Machine (FSM)* digunakan dalam penelitian ini sebagai pendekatan utama untuk mengatur alur kerja sistem gerbang tol otomatis secara terstruktur dan sistematis. FSM merupakan model komputasi yang membagi sistem ke dalam sejumlah *state* (keadaan), di mana setiap *state* merepresentasikan kondisi tertentu dan perpindahan antar *state* ditentukan oleh *input* atau kejadian yang terjadi [13].

Finite State Machine (FSM) pada sistem *embedded* dibangun atas beberapa komponen utama yang saling berinteraksi untuk mengendalikan perilaku sistem. Menurut Dash *et al.*, implementasi FSM terdiri atas *state*, *state transition*, *conditions*, dan *input events*, di mana setiap komponen memiliki peran yang berbeda dalam menentukan perubahan kondisi sistem [14]. *State* merupakan kondisi operasi yang sedang dijalankan oleh sistem pada suatu waktu. Setiap *state* merepresentasikan perilaku tertentu dan dapat menghasilkan aksi sesuai fungsi yang telah ditentukan. Selama sistem masih

berada pada state yang sama, sistem akan terus menjalankan perilaku tersebut hingga terdapat kondisi yang memungkinkan terjadinya perpindahan ke state berikutnya [14]. Pada penelitian ini, state direpresentasikan dalam beberapa kondisi operasi, seperti IDLE, WAIT_SENSOR1, VALIDATING, OPEN_GATE, dan CLOSE_GATE, yang masing-masing memiliki fungsi pengendalian *Automatic Line Barrier* (ALB).

State transition merupakan proses perpindahan sistem dari suatu *state* menuju *state* lainnya. Perpindahan tersebut tidak terjadi secara otomatis, melainkan harus dipicu oleh suatu kejadian (*event*) dan memenuhi kondisi (*condition*) yang telah ditentukan sehingga sistem dapat berpindah ke *state* berikutnya [14]. Pada penelitian ini, transisi *state* terjadi ketika sistem menerima masukan dari sensor ultrasonik, pembacaan RFID, maupun hasil validasi transaksi sehingga FSM dapat melanjutkan proses pengendalian ALB sesuai urutan operasi.

Condition merupakan sekumpulan aturan logika yang harus dipenuhi agar suatu *state transition* dapat dilakukan. Setiap *event* yang diterima sistem akan dievaluasi terlebih dahulu terhadap kondisi yang telah ditentukan. Apabila kondisi tersebut terpenuhi, maka sistem akan melakukan perpindahan menuju state berikutnya [14]. Dalam penelitian ini, contoh kondisi yang digunakan antara lain kendaraan terdeteksi oleh sensor ultrasonik, identifikasi RFID berhasil dilakukan, serta hasil validasi transaksi menunjukkan saldo kendaraan mencukupi.

Input event merupakan kejadian yang berasal dari lingkungan eksternal maupun dari proses internal sistem yang berfungsi sebagai pemicu perubahan perilaku sistem. *Event* tersebut akan memicu proses evaluasi kondisi, kemudian apabila seluruh kondisi telah terpenuhi maka sistem melakukan *state transition* menuju kondisi operasi berikutnya [14]. Pada penelitian ini, *input event* berasal dari pembacaan sensor ultrasonik, hasil identifikasi RFID, serta informasi hasil validasi transaksi yang diterima oleh mikrokontroler ESP32.

Pada sistem *embedded* berbasis *event-driven*, FSM bekerja dengan cara menunggu terjadinya suatu *input event* dari lingkungan. Ketika *event* diterima, sistem akan mengevaluasi kondisi (*condition*) yang telah ditentukan. Apabila

kondisi tersebut terpenuhi, maka sistem melakukan perpindahan (*state transition*) menuju *state* berikutnya dan menjalankan aksi (*action*) sesuai dengan *state* aktif. Proses ini berlangsung secara berulang selama sistem beroperasi sehingga alur kendali dapat berjalan secara sistematis dan terstruktur [15]. Pada penelitian ini, proses diawali dari *state* IDLE sebagai kondisi awal sistem. Selanjutnya, sensor ultrasonik dan RFID menghasilkan *event* yang akan dievaluasi oleh Firebase. Jika kondisi yang dipersyaratkan terpenuhi, sistem akan berpindah menuju *state* berikutnya hingga seluruh proses pengendalian *Automatic Line Barrier* selesai dilaksanakan.

Menurut IJECE (2015), implementasi FSM pada sistem *embedded* dapat dilakukan menggunakan pendekatan *event-driven*, di mana setiap *state* direpresentasikan dalam bentuk struktur program dan perpindahan *state* dikendalikan oleh *event* yang diterima sistem. Salah satu implementasi yang umum digunakan adalah struktur *nested switch statement*, sehingga proses pengendalian menjadi lebih terstruktur dan mudah dikembangkan [15]. Pada penelitian ini, setiap *state* direpresentasikan menggunakan enumerasi (*enum*), sedangkan perpindahan antar *state* dikendalikan menggunakan struktur *switch-case* berdasarkan *event* yang berasal dari sensor ultrasonik, identifikasi RFID, dan hasil validasi transaksi.

Finite State Machine banyak digunakan pada sistem *embedded* karena mampu menyederhanakan logika pengendalian yang kompleks menjadi sekumpulan *state* yang terstruktur. Pendekatan ini mempermudah proses implementasi, pengembangan, serta pengujian sistem karena setiap *state* dapat dirancang dan dianalisis secara terpisah [1]. Pada penelitian ini, penggunaan FSM memungkinkan setiap tahapan operasi *Automatic Line Barrier*, mulai dari pendeteksian kendaraan, validasi transaksi, pembukaan palang, hingga penutupan palang, dikendalikan secara berurutan sehingga alur kerja sistem menjadi lebih sistematis.

2.5 Sensor Ultrasonik HY-SRF05

Sensor ultrasonik HY-SRF05 merupakan sensor yang digunakan untuk mendeteksi keberadaan objek dengan cara mengukur jarak berdasarkan pantulan gelombang ultrasonik. Sensor ini bekerja menggunakan prinsip *time*

of flight (ToF), yaitu menghitung waktu yang diperlukan gelombang ultrasonik untuk merambat dari sensor menuju objek dan dipantulkan kembali ke sensor. Gelombang ultrasonik dipancarkan melalui transduser pemancar (*transmitter*) dengan frekuensi 40 kHz, kemudian diterima kembali oleh transduser penerima (*receiver*) setelah mengenai permukaan objek. Berdasarkan waktu tempuh gelombang tersebut, mikrokontroler menghitung jarak antara sensor dengan objek yang terdeteksi. Sensor ini banyak digunakan pada aplikasi robotika, sistem otomasi, *Internet of Things* (IoT), dan pembelajaran berbasis STEM karena mudah diintegrasikan dengan mikrokontroler Arduino maupun ESP32 [16], [17].



Gambar 2.3 Komponen sensor ultrasonik HY-SRF05

Berdasarkan Gambar 2.3, sensor HY-SRF05 terdiri atas dua buah *transduser* ultrasonik, yaitu *transmitter* dan *receiver*, serta lima buah pin antarmuka yang terdiri atas VCC, *Trigger*, *Echo*, *OUT*, dan GND. Pin *Trigger* digunakan untuk mengaktifkan proses pengiriman gelombang ultrasonik dengan memberikan pulsa logika *HIGH* selama sekitar 10 μ s, sedangkan pin *Echo* menghasilkan sinyal keluaran berupa durasi pulsa yang sebanding dengan waktu tempuh gelombang ultrasonik. Informasi tersebut kemudian diproses oleh mikrokontroler untuk memperoleh nilai jarak objek.

Pada penelitian ini, sensor HY-SRF05 digunakan sebagai sensor pendeteksi keberadaan kendaraan pada *prototype Automatic Line Barrier* (ALB). Sensor ditempatkan sebelum palang untuk mendeteksi kendaraan yang akan melintas serta ditempatkan di bawah palang yang digunakan sebagai sistem keamanan guna memastikan kendaraan telah melewati area gerbang sebelum palang ditutup kembali. Informasi jarak yang diperoleh sensor selanjutnya digunakan

sebagai masukan pada algoritma *Finite State Machine* (FSM) untuk menentukan perubahan keadaan (*state*) sistem sehingga proses buka dan tutup palang dapat berlangsung secara otomatis.

Tabel 2.2 Spesifikasi Sensor Ultrasonik HY-SRF05

Parameter	Nilai
Frekuensi ultrasonik	40 kHz
Tegangan kerja	4,5–5,5 V DC
Arus kerja	10–40 mA
Jarak pengukuran	2–450 cm
Resolusi	±0,3 cm
Akurasi	±3 mm
Sudut deteksi	≤15°
Frekuensi pengukuran	40 Hz
Trigger	Pulsa HIGH 10 μs
Jumlah pin	5 pin

Proses kerjanya dimulai dari pemancaran gelombang ultrasonik oleh transmitter, kemudian gelombang merambat di udara hingga mengenai objek, seperti kendaraan, dan dipantulkan kembali ke sensor dalam bentuk sinyal echo. Selanjutnya, sensor menghitung selisih waktu antara sinyal yang dipancarkan dan diterima untuk menentukan jarak objek dengan menggunakan prinsip *time of flight* [16].

Data yang diterima berupa durasi waktu dalam satuan mikrodetik (μs), yang kemudian dikonversi menjadi jarak dalam satuan sentimeter (cm) menggunakan persamaan:

$$d = \frac{v \times t}{2} \quad (2.1)$$

Keterangan:

d = Jarak (cm)

v = Kecepatan suara (0.0343 cm/μs)

t = Waktu (μs)

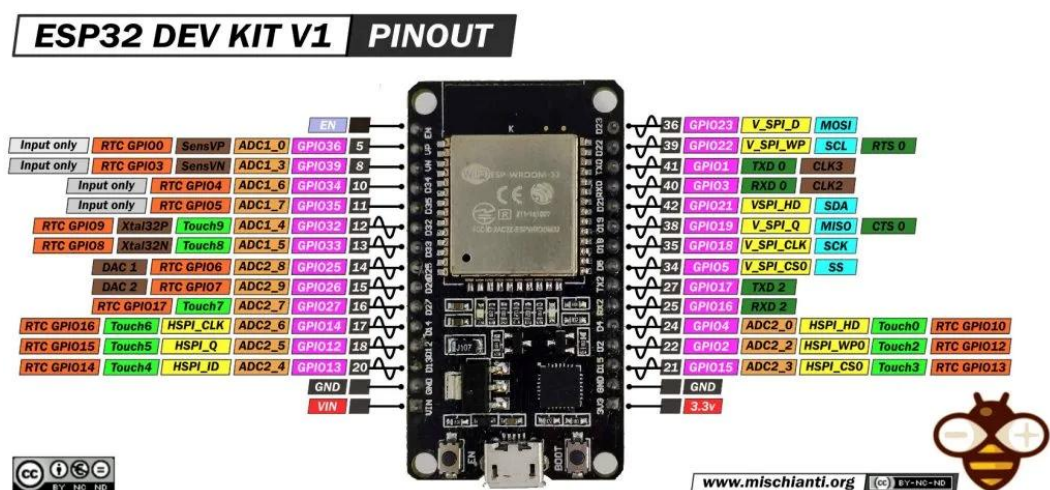
Faktor pembagi dua digunakan karena gelombang ultrasonik mengalami perjalanan bolak-balik dari sensor ke objek dan kembali ke sensor. Selanjutnya,

nilai jarak yang diperoleh dibandingkan dengan nilai ambang batas tertentu untuk menentukan keberadaan kendaraan. Perubahan nilai jarak yang signifikan digunakan sebagai indikator adanya kendaraan yang melintas.

Selain memiliki karakteristik pengukuran yang baik, penelitian Kusuma dan Santoso (2025) menunjukkan bahwa sensor HY-SRF05 memiliki *mean absolute error* sebesar 0,20 cm, persentase kesalahan sebesar 0,131%, serta tingkat keberhasilan deteksi sebesar 99,869%, sehingga sedikit lebih baik dibandingkan HC-SR04. Hasil tersebut menunjukkan bahwa HY-SRF05 layak digunakan pada sistem otomasi yang memerlukan pendeteksian objek secara presisi, termasuk pada sistem gerbang tol otomatis berbasis FSM [16].

2.6 Mikrokontroler ESP32

ESP32 merupakan mikrokontroler yang banyak digunakan dalam sistem otomasi berbasis *Internet of Things* (IoT) karena memiliki kemampuan pemrosesan data sekaligus komunikasi nirkabel dalam satu perangkat. Dalam sistem gerbang tol otomatis, ESP32 berperan sebagai pusat kendali yang mengintegrasikan berbagai komponen seperti sensor ultrasonik, sistem verifikasi transaksi, dan aktuator gerbang. Kemampuan ESP32 dalam mengolah data secara *real-time* memungkinkan sistem untuk merespons kondisi kendaraan yang terdeteksi dan menentukan aksi yang sesuai, seperti membuka atau menutup palang gerbang berdasarkan hasil verifikasi transaksi [18].



Gambar 2.4 Mikrokontroler ESP32 DevKit V1 [19]

Gambar 2.4 di atas merupakan mikrokontroler ESP32 beserta penjelasan semua pinoutnya. Dukungan konektivitas *WiFi* pada ESP32 memungkinkan perangkat untuk terhubung dengan *database* berbasis *cloud* seperti *Firebase*, sehingga proses pengambilan keputusan dapat dilakukan secara terintegrasi antara sistem lokal dan *server* [20]. Dengan demikian, penggunaan ESP32 dalam penelitian ini memberikan keunggulan dalam hal fleksibilitas, efisiensi, serta kemampuan integrasi sistem yang kompleks pada gerbang tol otomatis.

Tabel 2.3 Spesifikasi ESP32

Parameter	Nilai	Satuan / Keterangan
Prosesor	Dual-core Xtensa LX6	—
Kecepatan Clock	240	MHz
Memori RAM	520	KB
Flash Memory	4	MB
Tegangan Operasi	3.0 – 3.6	Volt (V)
GPIO	30	Pin
ADC	12	bit
Jumlah Channel ADC	15	Channel
DAC	2	Channel (8-bit)
PWM	Tersedia	Hampir semua pin
Interface	UART, SPI, I2C, I2S	—
Sensor Touch	10	Channel
Suhu Operasi	-40 hingga 125	°C
Konsumsi Daya	Rendah	Mendukung deep sleep

Pada Tabel 2.3 di atas merupakan spesifikasi yang ditawarkan dari mikrokontroler ESP32. Mulai dari tegangan operasi hingga jumlah pin yang tersedia.

2.7 Motor Servo

Motor servo pada penelitian ini digunakan sebagai aktuator utama yang berfungsi untuk menggerakkan palang gerbang tol secara otomatis berdasarkan kondisi sistem. Motor servo bekerja dengan prinsip kendali posisi menggunakan sinyal *Pulse Width Modulation* (PWM), sehingga sudut

pergerakan dapat diatur secara presisi sesuai dengan perintah dari mikrokontroler. Kemampuan tersebut menjadikan motor servo sangat sesuai digunakan pada sistem gerbang otomatis yang membutuhkan kontrol posisi buka dan tutup secara akurat.



Gambar 2.5 Motor servo MG996R [21]

Gambar 2.5 di atas merupakan penampakan dari motor servo MG996R. Dalam sistem yang dikembangkan, motor servo MG996R dikendalikan oleh ESP32 berdasarkan dua parameter utama, yaitu hasil deteksi kendaraan dari sensor ultrasonik dan hasil verifikasi transaksi dari sistem IoT (*Firebase*). Ketika kendaraan terdeteksi oleh sensor dan status transaksi dinyatakan valid (EPC terdaftar dan saldo mencukupi), maka ESP32 akan mengirimkan sinyal PWM untuk menggerakkan motor servo ke posisi terbuka. Sebaliknya, apabila kendaraan telah melewati area gerbang yang terdeteksi oleh sensor ultrasonik kedua, maka sistem akan memberikan perintah untuk mengembalikan servo ke posisi tertutup. Karakteristik MG996R yang memiliki torsi tinggi membuatnya mampu menggerakkan beban mekanik dengan lebih stabil dibandingkan servo mikro [22].

Tabel 2.4 Spesifikasi Motor Servo [21]

Parameter	Nilai	Satuan / Keterangan
Tegangan Operasi	4.8 – 7.2	Volt (V)
Torsi Maksimum	9.4 - 11	kg·cm
Kecepatan	±0.17	detik/60°
Sudut Rotasi	0 – 180	Derajat (°)

Parameter	Nilai	Satuan / Keterangan
Sinyal Kontrol	PWM	Pulse Width Modulation
Frekuensi PWM	50	Hz
Arus Operasi	500 - 900	mA
Material Gear	Metal Gear	-
Berat	±55	gram

Pada Tabel 2.4 [22], ditunjukkan karakteristik teknis dari motor servo MG996R yang digunakan sebagai aktuator penggerak ALB. Kemudian gaya yang bekerja pada palang diperoleh dari hasil perkalian massa palang dengan percepatan gravitasi sebagaimana ditunjukkan pada Persamaan (2.2).

$$F = m \times g \quad (2.2)$$

dengan:

$$F = \text{gaya (N)}$$

$$m = \text{massa palang (kg)}$$

$$g = \text{percepatan gravitasi (9,81 m/s}^2\text{)}$$

Selanjutnya, kebutuhan torsi minimum dihitung menggunakan Persamaan (2.3).

$$T = F \times r \quad (2.3)$$

dengan:

$$T = \text{torsi (N} \cdot \text{m)}$$

$$F = \text{gaya (N)}$$

$$r = \text{jarak titik berat palang terhadap poros servo (m)}$$

Untuk meningkatkan keandalan sistem, nilai torsi minimum dikalikan dengan *safety factor* sehingga diperoleh torsi perancangan. Pada penelitian ini digunakan *safety factor* sebesar 1,5–2 untuk mengantisipasi pengaruh gesekan, perubahan beban, serta percepatan awal saat servo mulai menggerakkan palang. Perhitungan torsi perancangan dinyatakan pada Persamaan (2.4).

$$T_{desain} = T \times SF \quad (2.4)$$

dengan:

$$T_{desain} = \text{torsi perancangan } (N \cdot m)$$

$$T = \text{torsi minimum } (N \cdot m)$$

$$SF = \text{Safety Factor}$$

Hasil perhitungan torsi umumnya dinyatakan dalam satuan *Newton-meter* ($N \cdot m$), sedangkan spesifikasi motor servo pada datasheet umumnya menggunakan satuan kilogram-sentimeter ($kg \cdot cm$). Oleh karena itu, diperlukan konversi satuan agar hasil perhitungan dapat dibandingkan secara langsung dengan spesifikasi motor servo yang digunakan. Hubungan antara kedua satuan tersebut dinyatakan pada Persamaan (2.5).

$$T_{kg \cdot cm} = \frac{T_{N \cdot m}}{0.09807} \quad (2.5)$$

dengan:

$$T_{kg \cdot cm} = \text{torsi dalam satuan kilogram – sentimeter } (kg \cdot cm)$$

$$T_{N \cdot m} = \text{torsi dalam satuan Newton – meter } (N \cdot m)$$

Konstanta 0,09807 merupakan nilai konversi antara satuan $kg \cdot cm$ dan $N \cdot m$, dengan hubungan yang dapat dilihat pada Persamaan (2.6):

$$1 kg \cdot cm = 0.09807 N \cdot m \quad (2.6)$$

2.8 Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan perangkat lunak yang digunakan sebagai lingkungan pengembangan untuk menulis, mengedit, mengompilasi, serta mengunggah program ke mikrokontroler. Arduino IDE bersifat *open-source* dan mendukung berbagai sistem operasi seperti Windows, macOS, dan Linux, sehingga banyak digunakan dalam pengembangan sistem *embedded* dan *Internet of Things* (IoT). Bahasa pemrograman yang digunakan mengacu pada bahasa C/C++ yang telah disederhanakan melalui berbagai *library*, sehingga memudahkan pengguna dalam mengontrol perangkat keras seperti sensor dan aktuator [23].



Gambar 2.6 Arduino IDE [24]

Pada **Gambar 2.7** merupakan tampilan *loading screen* dari perangkat lunak Arduino IDE. Dalam penggunaannya, Arduino IDE menyediakan fitur utama berupa *code editor*, *compiler*, dan *uploader* yang memungkinkan program (*sketch*) dapat langsung diunggah ke mikrokontroler. Selain itu, ketersediaan *library* dan contoh program mendukung proses pengembangan sistem menjadi lebih efisien dan terstruktur [25]. Pada penelitian ini, Arduino IDE digunakan sebagai perangkat lunak untuk mengimplementasikan logika kendali berbasis *Finite State Machine* (FSM) pada mikrokontroler ESP32, sehingga proses pengolahan data dari sensor serta pengendalian aktuator dapat dilakukan secara sistematis dan *real-time*.

2.9 Power Supply 12V 5A

Catu daya (*power supply*) merupakan komponen utama dalam sistem elektronika yang berfungsi untuk menyediakan energi listrik dengan tegangan dan arus yang stabil sesuai kebutuhan perangkat. Dalam sistem berbasis mikrokontroler dan otomasi, kestabilan tegangan sangat penting karena berpengaruh langsung terhadap kinerja sensor, aktuator, serta komunikasi data [26].



Gambar 2.7 *Power Supply 12V 5A* [27]

Berdasarkan **Gambar 2.7**, *Power Supply Unit* (PSU) yang digunakan dalam penelitian ini adalah catu daya dengan spesifikasi 12V 5A, yang berfungsi sebagai sumber tegangan utama untuk menyuplai seluruh komponen sistem, seperti mikrokontroler, sensor ultrasonik HY-SRF05, modul RFID, LCD, *pilot lamp* serta aktuator berupa motor servo. Pemilihan PSU dengan arus sebesar 5A bertujuan untuk memastikan ketersediaan daya yang stabil dan mencukupi, terutama ketika beberapa komponen bekerja secara bersamaan. Tegangan 12V dipilih karena sesuai dengan kebutuhan operasional sebagian besar komponen elektronik yang digunakan dalam sistem.

Power supply yang digunakan menerapkan teknologi *Switch Mode Power Supply* (SMPS). Dibandingkan catu daya linier, SMPS memiliki efisiensi yang lebih tinggi, ukuran yang lebih ringkas, serta mampu menghasilkan tegangan keluaran yang relatif stabil meskipun terjadi perubahan beban. Oleh karena itu, SMPS banyak digunakan pada sistem kendali industri, sistem IoT, dan perangkat elektronika modern yang membutuhkan efisiensi energi serta keandalan operasi .

Tabel 2.5 Spesifikasi *Power Supply 12V 5A* [27]

Parameter	Spesifikasi
Input Tegangan	220 VAC
Tegangan Output	12 VDC
Arus Output Maks	5 A
Daya Output	60 W

Parameter	Spesifikasi
Frekuensi Switching	50 – 150 kHz
Efisiensi	85 – 92 %
Topologi	Flyback / Forward SMPS
Proteksi	Overcurrent, Overvoltage, Short Circuit

Pada **Tabel 2.5**, ditunjukkan spesifikasi dari komponen *power supply* 12 VDC 5A yang digunakan sebagai sumber daya utama dalam sistem. Pada penelitian ini, digunakan *power supply* 12V 5A sebagai sumber tegangan utama yang mendistribusikan daya ke seluruh sistem. Tegangan 12V digunakan untuk menyuplai beban utama, kemudian diturunkan menjadi 12V menggunakan *buck converter* untuk memenuhi kebutuhan mikrokontroler ESP32, sensor ultrasonik, serta modul komunikasi IoT. Konsep distribusi tegangan ini umum digunakan pada sistem berbasis mikrokontroler untuk menyesuaikan kebutuhan tegangan dari berbagai komponen dalam satu sistem [28].

Setiap komponen memiliki kebutuhan arus yang berbeda sesuai dengan karakteristik dan fungsi masing-masing. Untuk menentukan spesifikasi catu daya yang sesuai, terlebih dahulu diperlukan perhitungan total arus yang dibutuhkan oleh seluruh komponen. Total arus sistem diperoleh dengan menjumlahkan arus kerja setiap komponen elektronik yang beroperasi secara bersamaan. Persamaan total arus sistem ditunjukkan pada Persamaan (2.7).

$$I_{total} = \sum_{i=1}^n I_i \quad (2.7)$$

Keterangan:

I_{total} = total arus sistem (A)

I_i = arus masing – masing komponen ke – i (A)

n = jumlah komponen pada sistem

Setelah total arus sistem diperoleh, langkah berikutnya adalah menghitung kebutuhan daya listrik. Daya listrik merupakan besarnya energi listrik yang

digunakan oleh suatu beban dalam setiap satuan waktu dan dipengaruhi oleh besar tegangan serta arus yang mengalir pada komponen tersebut. Hubungan antara daya, tegangan, dan arus dinyatakan menggunakan **Persamaan (2.8)**.

$$P = V \times I \quad (2.8)$$

Keterangan:

$P = \text{daya listrik } (W)$

$V = \text{tegangan kerja } (V)$

$I = \text{arus listrik } (A)$

Pada penelitian ini sistem catu daya dibagi menjadi beberapa jalur menggunakan buck converter sesuai dengan kebutuhan tegangan masing-masing komponen. Oleh karena itu, kebutuhan daya total sistem dihitung dengan menjumlahkan daya pada setiap jalur catu daya. Perhitungan ini bertujuan untuk memperoleh kebutuhan daya keseluruhan yang harus mampu disuplai oleh power supply utama. Persamaan total kebutuhan daya sistem ditunjukkan pada Persamaan (2.9).

$$P_{total} = \sum_{i=1}^n P_i \quad (2.9)$$

Keterangan

$P_{total} = \text{total kebutuhan daya sistem } (W)$

$P_i = \text{daya pada jalur catu daya ke } - i \text{ } (W)$

$n = \text{jumlah jalur catu daya}$

Dalam perancangan sistem elektronika, kapasitas catu daya tidak disarankan dipilih sama dengan hasil perhitungan kebutuhan daya teoritis. Hal tersebut bertujuan untuk mengantisipasi adanya lonjakan arus (*current surge*), toleransi spesifikasi komponen, serta kemungkinan penambahan beban pada pengembangan sistem di masa mendatang. Oleh karena itu, digunakan *Safety Factor* (SF) sehingga kapasitas catu daya yang dipilih memiliki cadangan daya yang memadai. Kebutuhan daya rancangan dihitung menggunakan Persamaan (2.10).

$$P_{design} = P_{total} \times SF \quad (2.10)$$

Keterangan

P_{design} = kebutuhan daya rancangan (W)

P_{total} = total kebutuhan daya sistem (W)

SF = Safety Factor

2.10 Modul LM2596

Modul LM2596 merupakan konverter DC-DC tipe *step-down* (*buck converter*) yang digunakan untuk menurunkan DC dari nilai yang lebih tinggi menjadi tegangan yang lebih rendah. Dibandingkan regulator linear, LM2596 memiliki efisiensi yang lebih tinggi karena menggunakan metode *switching* dalam proses konversi tegangan, sehingga energi yang hilang dalam bentuk panas menjadi lebih kecil [29].



Gambar 4.1 Buck converter LM2596 [30]

Modul LM2596 pada **Gambar 4.1** banyak digunakan pada berbagai sistem elektronika karena mampu menghasilkan tegangan keluaran yang stabil dengan kemampuan menyediakan arus yang cukup besar. Karakteristik tersebut menjadikan LM2596 sesuai digunakan sebagai catu daya pada sistem berbasis mikrokontroler, sensor, maupun modul komunikasi [31], [32].

Tabel 2.6 Spesifikasi modul LM2596 [30]

Parameter	Spesifikasi
Tipe konverter	<i>Buck Converter (Step-Down)</i>
IC utama	LM2596
Tegangan masukan	3,2 – 40 V DC
Tegangan keluaran	1,25 – 35 V DC (<i>Adjustable</i>)
Arus keluaran maksimum	2 A
Frekuensi <i>switching</i>	150 kHz

Efisiensi	Hingga $\pm 90\%$
-----------	-------------------

Berdasarkan Tabel 2.6, modul LM2596 memiliki spesifikasi yang sesuai untuk digunakan sebagai catu daya pada sistem elektronika yang memerlukan tegangan DC stabil. Rentang tegangan masukan yang cukup lebar, yaitu 4,5V hingga 40V, memungkinkan modul ini digunakan bersama berbagai sumber catu daya, sedangkan kemampuan menghasilkan arus keluaran hingga 3A menjadikannya mampu menyuplai beberapa perangkat secara bersamaan [29].

Pada penelitian ini, spesifikasi tersebut dimanfaatkan untuk menurunkan tegangan keluaran *power supply* 12V menjadi 5V sebagai sumber catu daya bagi ESP32 dan tiga sensor ultrasonik HY-SRF05. Dengan demikian, seluruh komponen memperoleh tegangan kerja yang sesuai sehingga proses pendeteksian kendaraan dan pengendalian *Automatic Line Barrier* berbasis *Finite State Machine* dapat berlangsung.

2.11 Modul XL4005

Modul XL4005 merupakan modul yang digunakan untuk menurunkan tegangan DC dari nilai yang lebih tinggi menjadi tegangan yang lebih rendah. Modul ini bekerja menggunakan prinsip *buck converter*, yaitu mengubah tegangan masukan menjadi tegangan keluaran yang lebih rendah dengan efisiensi yang tinggi. Oleh karena itu, modul XL4005 banyak digunakan sebagai catu daya pada berbagai perangkat elektronika yang membutuhkan tegangan DC yang stabil [33], [34]. Berdasarkan *datasheet* XLSEMI, modul yang menggunakan IC XL4005 memiliki tegangan masukan 5V hingga 32V dan tegangan keluaran yang dapat diatur mulai 0,8V hingga 30V.



Gambar 2.8 Buck converter XL4005 [35]

Modul ini juga mampu menghasilkan arus keluaran hingga 5A, serta dilengkapi dengan fitur *thermal protection* dan *current limiting* untuk melindungi rangkaian dari suhu maupun arus yang berlebihan [33].

Tabel 2.7 Spesifikasi modul XL4005 [35]

Parameter	Spesifikasi
Jenis modul	DC-DC Buck Converter (Step-Down)
IC utama	XL4005
Tegangan masukan	4–35 V DC
Tegangan keluaran	1,25–32 V DC (Adjustable)
Arus keluaran maksimum	5 A
Frekuensi <i>switching</i>	300 kHz
Suhu Operasional	-40°C s/d +85°C

Berdasarkan **Tabel 2.7**, modul XL4005 memiliki spesifikasi yang sesuai untuk digunakan sebagai catu daya motor servo MG996R. Modul ini dapat menerima tegangan masukan hingga 35V dan menghasilkan tegangan keluaran yang dapat diatur sesuai kebutuhan. Selain itu, kemampuan menghasilkan arus hingga 5A membuat modul ini mampu menyuplai daya yang dibutuhkan servo saat membuka maupun menutup palang. Fitur *thermal protection* dan *current limiting* juga membantu menjaga modul tetap bekerja dengan aman selama sistem beroperasi [33].

Modul XL4005 dipilih karena mampu menyediakan tegangan 6V yang sesuai dengan kebutuhan kerja motor servo MG996R. Selain itu, kemampuan modul dalam menghasilkan arus yang cukup besar membuat servo dapat bekerja secara optimal ketika membuka dan menutup palang. Penggunaan modul XL4005 secara terpisah dari catu daya ESP32 dan sensor ultrasonik juga membantu menjaga kestabilan tegangan pada sistem kendali, sehingga seluruh komponen dapat bekerja dengan baik selama proses identifikasi kendaraan dan pengoperasian *Automatic Line Barrier*.