

**PENGEMBANGAN SISTEM KONTROL PINTU OTOMATIS  
DAN PERGERAKAN ROBOT BERBASIS *INTERNET OF  
THINGS(IoT)* PADA ROBOT PENGANTAR MAKANAN  
(*DEVELOPMENT OF AN AUTOMATIC DOOR CONTROL  
SYSTEM AND ROBOT MOVEMENT BASED ON THE  
INTERNET OF THINGS (IoT) ON FOOD DELIVERY ROBOTS*)**

**BUKU TUGAS AKHIR**

Disusun sebagai syarat mata kuliah Tugas Akhir

Program Studi S1 Teknik Elektro

Disusun oleh

**NIDYA INTANI DWI SAKA PUTRI**

**1102213023**



**Universitas  
Telkom**

**FAKULTAS TEKNIK ELEKTRO**

**UNIVERSITAS TELKOM**

**BANDUNG**

**2025**

# **LEMBAR PENGESAHAN**

## **TUGAS AKHIR**

### **PENGEMBANGAN SISTEM KONTROL PINTU OTOMATIS DAN PERGERAKAN ROBOT BERBASIS *INTERNET OF THINGS(IoT)* PADA ROBOT PENGANTAR MAKANAN**

**(DEVELOPMENT OF AN AUTOMATIC DOOR CONTROL SYSTEM  
AND ROBOT MOVEMENT BASED ON THE INTERNET OF THINGS  
(IoT) ON FOOD DELIVERY ROBOTS)**

Telah disetujui dan disahkan untuk mengikuti sidang Tugas Akhir

**Program Studi Teknik Elektro**

**Fakultas Teknik Elektro**

**Universitas Telkom**

**Disusun oleh:**

**NIDYA INTANI DWI SAKA PUTRI**

**1102213023**

**Bandung, tanggal 15 Juli 2025**

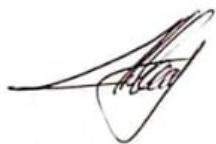
**Pembimbing I**



**Dr. ERWIN SUSANTO, S.T., M.T.**

**NIP.07740045**

**Pembimbing II**



**IRHAM MULKAN RODIANA, S.T., M.T.**

**NIP. 20950004**

**Dosen Pengusul Topik**



**Dr. ERWIN SUSANTO, S.T., M.T.**

**NIP.07740045**

## **ABSTRAK**

Perkembangan teknologi robotika memberikan peluang untuk meningkatkan efisiensi dan kualitas layanan, salah satunya melalui robot pengantar makanan dengan kemampuan membuka pintu otomatis. Sistem ini dirancang untuk mengatasi masalah kebersihan, efisiensi, dan akurasi dalam pengantaran makanan. Penelitian ini bertujuan mengembangkan sistem kontrol pintu otomatis dan monitoring berbasis Internet of Things (IoT) untuk efisiensi kontrol pintu otomatis.

Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem pintu otomatis berbasis kontrol PID dengan menggunakan rotary encoder sebagai sensor posisi. Limit switch digunakan sebagai pengaman tambahan agar pintu dapat berhenti tepat di batas akhir dan mencegah kerusakan mekanik. Selain itu, sistem dilengkapi komunikasi dua arah antara modul kontrol, serta integrasi teknologi Internet of Things (IoT) agar status pintu dapat dipantau dan dikendalikan secara jarak jauh.

Hasil pengujian menunjukkan bahwa kombinasi parameter kontrol PID yang optimal dapat menghasilkan gerakan pintu yang presisi. Sistem ini juga terbukti stabil dan responsif, sehingga mampu mendukung proses pengantaran makanan dengan aman, efisien, dan tetap menjaga kualitas makanan agar higienis. Dengan adanya pintu otomatis yang terintegrasi dengan IoT, robot pengantar makanan menjadi lebih inovatif, dan memberikan kemudahan serta kenyamanan tambahan bagi pengguna.

**Kata Kunci:** IoT, kontrol PID, pintu otomatis, rotary encoder, Robot pengantar makanan.

## **ABSTRACT**

The advancement of robotics technology offers opportunities to improve service efficiency and quality, one of which is through food delivery robots equipped with automatic door capabilities. This system is designed to address issues of hygiene, efficiency, and accuracy in the food delivery process. This study aims to develop an automatic door control and monitoring system based on the Internet of Things (IoT) to improve the efficiency of automatic door operations.

This research focuses on designing and implementing an automatic door system using PID control with a rotary encoder as a position feedback sensor. A limit switch is used as an additional safety feature to ensure the door stops precisely at the end limit and prevents mechanical damage. Furthermore, the system is equipped with two-way communication between control modules and integrated with IoT technology, allowing the door status to be monitored and controlled remotely.

Test results show that the optimal combination of PID control parameters produces precise door movements. The system has also proven to be stable and responsive, thus supporting the food delivery process in a safe and efficient manner while maintaining the food's hygienic quality. With the integration of the automatic door and IoT, the food delivery robot becomes more innovative and provides additional convenience and comfort for users.

**Keywords:** IoT, PID control, automatic door, rotary encoder, food delivery robot.

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas segala rahmat, karunia, dan kemudahan yang telah diberikan sehingga penulis dapat menyelesaikan penyusunan tugas akhir yang berjudul "**Pengembangan Sistem Kontrol Dan Monitoring Pintu Otomatis Berbasis Internet Of Things (IoT) Pada Robot Pengantar Makanan**" dengan lancar.

Penulisan tugas akhir ini disusun sebagai salah satu syarat dalam menyelesaikan studi pada program Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik Elektro, Telkom University. Dalam proses penyusunan Tugas Akhir ini, penulis menyadari bahwa tanpa bantuan dan dukungan dari berbagai pihak, penyusunan tugas akhir ini tidak akan berjalan dengan lancar..

Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu, baik secara langsung maupun tidak langsung, dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan di masa yang akan datang. Semoga Tugas Akhir ini dapat memberikan manfaat, baik bagi penulis sendiri, maupun bagi pembaca dan pihak-pihak yang berkepentingan.

## UCAPAN TERIMA KASIH

Dalam proses penyusunan Tugas Akhir ini, penulis menerima berbagai bentuk bantuan yang sangat berarti, baik dukungan moral, bimbingan akademik, maupun saran yang membangun dari berbagai pihak. Oleh karena itu dengan penuh rasa hormat dan tulus, penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah berkontribusi dalam proses penyusunan Tugas Akhir ini:

1. Allah SWT, atas segala nikmat, kekuatan, dan kesehatan yang telah diberikan.
2. Orang tua tercinta, atas doa, dukungan moral, dan semangat yang diberikan serta motivasi yang tiada henti-hentinya selalu diberikan kepada penulis selama melakukan masa studi hingga penyusunan Tugas Akhir ini.
3. Bapak Dr. Erwin Susanto, S.T., M.T. dan Bapak Irham Mulkan Rodiana, S.T., M.T. selaku Dosen pembimbing pertama dan kedua, yang telah dengan sabar membimbing, mengarahkan, dan memberikan masukan dalam setiap tahap penyusunan Tugas Akhir ini.
4. Teman-teman tim T.I.F.A yang telah banyak membantu, mendukung, dan menemani penulis selama proses penggerjaan Tugas Akhir ini.
5. Teman-teman Keluarga Besar Kang Gabut, atas canda tawa, kebersamaan, dan dukungan emosional yang sangat membantu dalam menjaga semangat penulis.
6. Teman-teman RL Laboratory, atas bantuan teknis dan kerja sama selama menempuh studi di lingkungan laboratorium.
7. Teman-teman kelas EL-45-04, atas kebersamaan, kerja sama, dan semangat belajar yang senantiasa membantu penulis untuk terus maju
8. Giovanni Salim, atas dukungan dan teman projek dalam penggerjaan Tugas Akhir.
9. Seluruh dosen Program Studi S1 Teknik Elektro Telkom University, atas ilmu, wawasan, dan dedikasi dalam proses pembelajaran selama ini.
10. Dan semua pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan bantuan maupun dukungan selama proses penyusunan Tugas Akhir ini.

Tanpa kehadiran, bimbingan, serta dukungan dari semua pihak di atas, tentu Tugas Akhir ini tidak akan dapat terselesaikan dengan baik. Semoga segala bantuan dan kebaikan yang telah diberikan menjadi amal yang dibalas dengan kebaikan yang berlipat ganda.

## DAFTAR ISI

<b>LEMBAR PENGESAHAN .....</b>	<b>ii</b>
<b>ABSTRAK .....</b>	<b>iii</b>
<b>ABSTRACT .....</b>	<b>iv</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>DAFTAR ISI.....</b>	<b>vii</b>
<b>DAFTAR GAMBAR.....</b>	<b>ix</b>
<b>DAFTAR TABEL .....</b>	<b>x</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1    Latar Belakang Masalah.....	1
1.2    Rumusan Masalah.....	2
1.3    Tujuan .....	2
1.4    Manfaat Hasil Penelitian .....	2
1.5    Batasan Masalah.....	3
1.6    Metode Penelitian .....	3
1.7    Proyeksi Pengguna .....	4
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>5</b>
2.1    Internet of Things (IoT) .....	5
2.1.1    Arsitektur IoT .....	5
2.1.2    Kodular .....	6
2.2    Robotika .....	7
2.2.1    BELLABOT.....	7
2.2.2    Sistem Pintu Geser Otomatis .....	8
2.3    Mekanisme Lazy Susan.....	8
2.4    Sistem Kontrol .....	9
2.4.1    Open Loop Control (Sistem Kontrol Terbuka).....	9
2.4.2    Closed Loop (Sistem Kontrol Tertutup) .....	9
2.5    Kontrol PID.....	10
2.4.3    Respon Transient .....	11
2.6    Rotary Encoder.....	13

<b>2.7</b>	<b>Penelitian Terdahulu.....</b>	<b>14</b>
<b>BAB III PERANCANGAN SISTEM .....</b>		<b>16</b>
<b>3.1</b>	<b>Desain Sistem .....</b>	<b>16</b>
<b>3.1.1</b>	<b>Diagram Blok.....</b>	<b>16</b>
<b>3.1.2</b>	<b>Fungsi dan Fitur .....</b>	<b>18</b>
<b>3.2</b>	<b>Desain Perangkat Keras .....</b>	<b>19</b>
<b>3.2.1</b>	<b>Desain Pintu.....</b>	<b>19</b>
<b>3.2.2</b>	<b>Desain UI.....</b>	<b>20</b>
<b>3.2.3</b>	<b>Diagram Kontroler Pintu Otomatis .....</b>	<b>21</b>
<b>3.2.4</b>	<b>Diagram Blok Perangkat Keras.....</b>	<b>22</b>
<b>3.2.5</b>	<b>Desain PCB Mikrokontroler .....</b>	<b>23</b>
<b>3.2.6</b>	<b>Spesifikasi Komponen.....</b>	<b>23</b>
<b>3.3</b>	<b>Desain Perangkat Lunak .....</b>	<b>32</b>
<b>3.3.1</b>	<b>Spesifikasi Sub Sistem Pintu dan <i>Tray</i>.....</b>	<b>33</b>
<b>BAB IV HASIL DAN ANALISIS .....</b>		<b>35</b>
<b>4.1</b>	<b>Hasil Percobaan.....</b>	<b>35</b>
<b>4.1.1</b>	<b>Pengujian Motor DC JGY370 dan Driver BTS7960 .....</b>	<b>35</b>
<b>4.1.2</b>	<b>Pengujian Limit Switch .....</b>	<b>37</b>
<b>4.1.3</b>	<b>Pengujian Rotary Encoder .....</b>	<b>38</b>
<b>4.1.4</b>	<b>Pengujian Kontrol PID Pintu.....</b>	<b>40</b>
<b>4.1.5</b>	<b>Pengujian Sistem Jarak Jauh (Internet of Things).....</b>	<b>47</b>
<b>4.1.6</b>	<b>Pengujian Keamanan Makanan .....</b>	<b>48</b>
<b>4.2</b>	<b>Analisis .....</b>	<b>50</b>
<b>4.2.1</b>	<b>Analisis fungsi <i>Limit Switch</i> KW10-ZIP sebagai Pengaman dan Pengontrol Posisi.....</b>	<b>50</b>
<b>4.2.2</b>	<b>Analisis Integrasi Sistem Monitoring dan Kontrol IoT.....</b>	<b>51</b>
<b>4.2.3</b>	<b>Analisis Penggunaan Kontrol PID Pada Pergerakan Pintu.....</b>	<b>52</b>
<b>4.2.4</b>	<b>Analisis Pengujian Keamanan Makanan.....</b>	<b>54</b>
<b>BAB V KESIMPULAN DAN SARAN .....</b>		<b>55</b>
<b>5.1</b>	<b>Kesimpulan .....</b>	<b>55</b>
<b>5.2</b>	<b>Saran.....</b>	<b>56</b>
<b>DAFTAR PUSTAKA.....</b>		<b>57</b>
<b>LAMPIRAN.....</b>		<b>60</b>

## DAFTAR GAMBAR

Gambar 2. 1 Robot BELLABOT .....	8
Gambar 2. 2 Diagram Blok Kontrol PID .....	11
Gambar 2. 3 Bentuk Sinyal Tanggapan Transien .....	12
Gambar 3. 1 Diagram Blok Sistem pintu Otomatis .....	16
Gambar 3. 2 3D Desain Pintu .....	19
Gambar 3. 3 Desain UI Tampilan Layar 2 .....	20
Gambar 3. 4 Desain UI Tampilan Layar 1 .....	20
Gambar 3. 6 Diagram Kontroler Pintu Otomatis .....	21
Gambar 3. 7 Diagram Blok Perangkat Keras.....	22
Gambar 3. 8 Desain PCB Mikrokontroler .....	23
Gambar 3. 9 Limit Switch Microswitch KW10-Z1P .....	23
Gambar 3. 10 BTS7960 .....	24
Gambar 3. 11 Baterai Lithium Ion 3000 mAh 48V .....	25
Gambar 3. 12 Stepdown LM2596.....	27
Gambar 3. 13 ESP32 DEV-KIT.....	28
Gambar 3. 14 Rotary Encoder E38S6G5 600B G24N.....	29
Gambar 3. 15 Motor GearBox DC JGY 370.....	30
Gambar 3. 16 Raspberry Pi 4 .....	31
Gambar 4. 1 Grafik saat Rotary Encoder Searah Jarum Jam.....	39
Gambar 4. 2 Grafik saat Rotary Encoder Lawan Arah Jarum Jam.....	40
Gambar 4. 3 Grafik saat Pintu Buka .....	42
Gambar 4. 4 Grafik saat Pintu Tutup .....	42
Gambar 4. 5 Grafik saat Pintu Buka (2).....	43
Gambar 4. 6 Grafik saat Pintu Tutup (2).....	43
Gambar 4. 7 Grafik saat Pintu Buka (3).....	44
Gambar 4. 8 Grafik saat Pintu Tutup (3).....	44
Gambar 4. 9 Grafik saat Pintu Buka (4).....	45
Gambar 4. 10 Grafik saat Pintu Tutup (4) .....	45
Gambar 4. 11 Grafik saat Pintu Buka (5).....	46
Gambar 4. 12 Grafik saat Pintu Tutup (5) .....	46
Gambar 4. 13 Pengujian Sistem IoT .....	48
Gambar 4. 14 Tampilan IoT Keamanan Makanan.....	50

## **DAFTAR TABEL**

Tabel 3. 1 Perbandingan Limit Switch.....	24
Tabel 3. 2 Perbandingan Motor Driver .....	24
Tabel 3. 3 Spesifikasi Baterai Lithium Ion 3000 mAh 48V .....	26
Tabel 3. 4 Perbandingan Stepdown.....	27
Tabel 3. 5 Perbandingan mikrokontroler utama.....	28
Tabel 3. 6 Perbandingan Sensor Rotary .....	29
Tabel 3. 7 Spesifikasi Motor GearBox DC .....	30
Tabel 3. 8 Spesifikasi Raspberry Pi 4 .....	31
Tabel 4. 1 Pengujian Motor DC dan Driver BTS7960.....	36
Tabel 4. 2 Pengujian nilai arus motor DC jGY370.....	36
Tabel 4. 3 Pengujian Limit Switch KW10-ZIP.....	37
Tabel 4. 4 Pengujian Sensor rotary encoder.....	39
Tabel 4. 5 Pengujian Delay Sistem .....	47
Tabel 4. 6 Pengujian delay sistem keamanan pintu .....	49

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Pesatnya perkembangan teknologi robotika telah meningkatkan peran robot dalam menunjang aktivitas manusia, termasuk di sektor pelayanan kafe dan restoran [1]. Berbagai inovasi kini memungkinkan terciptanya solusi yang mempermudah operasional bisnis kuliner, salah satunya adalah pengembangan robot pengantar makanan dengan kemampuan membuka pintu secara otomatis. Teknologi ini diharapkan dapat meningkatkan efisiensi pelayanan sekaligus menjaga kebersihan serta kualitas makanan yang disajikan kepada pelanggan [2].

Dalam operasional kafe, penggunaan tenaga manusia untuk mengantar makanan sering kali menghadapi beberapa kendala, seperti keterbatasan tenaga kerja, potensi kesalahan dalam pengantaran, serta masalah kebersihan karena adanya kontak langsung dengan lingkungan sekitar. Hal ini dapat berdampak pada kepuasan pelanggan sekaligus meningkatkan biaya operasional bagi pemilik kafe [3]. Selain itu, diperlukan sistem keamanan pada robot pengantar makanan agar pelanggan atau orang lain yang tidak berhak tidak dapat mengambil makanan sebelum sampai kepada pemesan. Dengan demikian, keamanan makanan tetap terjaga hingga diterima oleh pelanggan yang tepat [1].

Untuk mengatasi tantangan tersebut, perlu dilakukan penelitian mengenai pemilihan komponen mekanik yang tepat, terutama untuk sistem pintu otomatis pada robot pengantar makanan. Sistem ini dirancang menggunakan mekanisme lazy susan yang dilengkapi dengan worm gear agar pintu dapat bergerak memutar dengan konsisten dan halus, sesuai dengan kebutuhan pelayanan di lingkungan kafe yang dinamis.

Penelitian sebelumnya menunjukkan bahwa masih terdapat kekurangan pada desain pintu otomatis, seperti belum optimalnya mekanisme pergerakan,

desain pintu yang tidak menyesuaikan kebutuhan, serta kontrol yang belum sepenuhnya otomatis. Hal ini menyebabkan sistem pintu belum mampu memberikan perlindungan maksimal terhadap makanan dari paparan lingkungan, sehingga kebersihan dan sterilisasi belum optimal [2].

Berdasarkan permasalahan tersebut, diperlukan pengembangan pintu otomatis yang mampu bekerja secara efisien, presisi, serta mendukung standar kebersihan di berbagai lingkungan. Sistem ini juga harus dilengkapi dengan sensor yang sesuai dan kontrol yang baik agar dapat beroperasi secara maksimal tanpa memerlukan campur tangan manusia secara langsung.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, dapat dirumuskan masalah sebagai berikut:

1. Bagaimana cara mengoptimalkan sistem kontrol robot agar dapat bergerak secara baik/konsisten dan menghindari berbagai objek penghalang?
2. Bagaimana meningkatkan efisiensi monitoring dan kontrol terhadap pergerakan pada pintu?

## 1.3 Tujuan

Tujuan yang hendak dicapai dari penelitian Tugas Akhir ini adalah

1. Robot mampu mengantarkan dan menghidangkan makanan dengan aman dan steril kepada pelanggan dengan kemampuan sistem mekatronik yang baik. Dengan demikian proses pengantaran makanan pada restoran, kafe atau kantin akan lebih efektif dan efisien.
2. Mampu mengembangkan sistem kontrol pintu otomatis menggunakan kontrol PID dan dapat memonitoring pintu berbasis *Internet of Things* (IoT), sehingga dapat meningkatkan efisiensi monitoring dan kontrol pintu.

## 1.4 Manfaat Hasil Penelitian

Dari penelitian ini, diharapkan dapat memberikan manfaat, diantaranya:

1. Efisiensi:
  - Penghematan waktu robot pengantar makanan pada proses membuka dan menutup pintu makanan menjadi lebih cepat dan otomatis.
  - Tenaga manusia dapat dialokasikan untuk tugas-tugas lain. Ini juga dapat meningkatkan kecepatan pengantaran makanan.
2. Keamanan:
  - Mengurangi risiko terjepit atau terbentur pintu dapat diminimalkan.
  - Mengurangi interaksi langsung antara staf dan pelanggan dapat membantu mengurangi risiko penyebaran penyakit, terutama di restoran yang sedang mengalami peningkatan pengunjung.
  - Makanan yang diantar oleh robot akan lebih terjaga kebersihannya.
3. Kenyamanan:

Pengguna dapat menikmati layanan pengiriman makanan yang lebih nyaman dan efisien. Serta biaya tenaga kerja dapat dikurangi.

## 1.5 Batasan Masalah

Batasan Masalah pada penelitian ini adalah sebagai berikut:

1. Pintu robot pengantar makanan dapat bergerak secara otomatis.
2. Pintu berbentuk setengah lingkaran dengan diameter  $\pm 57$  cm.
3. Penelitian ini berfokus pada pengembangan pergerakan robot dan sistem kontrol pintu otomatis berbasis *Internet of Things* (IoT).
4. Pergerakan robot terbatas pada area dalam ruangan.
5. Data yang dimonitor terbatas pada status pintu.
6. Pengujian akan dibatasi pada lingkungan Tel-U Coffee.

## 1.6 Metode Penelitian

Pada penyusunan Tugas Akhir ini metode penelitian yang digunakan adalah sebagai berikut:

a. Studi Literatur

Tahap studi literatur adalah saat penulis meneliti dan menganalisis teori-teori yang diperlukan untuk menyusun Tugas Akhir. Sumber yang

digunakan mencakup buku, jurnal, laporan penelitian yang diterbitkan secara nasional dan internasional, serta berbagai sumber di internet dan hasil diskusi yang telah dilakukan.

b. Perancangan Alat

Dalam tahap ini, dilakukan pemodelan mekanik, elektronik, dan algoritma, serta pengembangan sistem kontrol untuk sistem mekatronik pintu otomatis dan kontrol robot.

c. Simulasi dan Implementasi

Tahap ini melibatkan penggabungan semua desain yang telah dibuat menjadi sebuah perangkat keras. Hasil dari tahap ini diharapkan memenuhi target yang telah ditentukan.

d. Pengujian Alat

Setelah menjadi perangkat keras, alat tersebut akan diuji untuk mengevaluasi kinerja sistem yang telah dibangun.

e. Analisis dan Evaluasi

Setelah pengujian alat, dilakukan analisis dan evaluasi terhadap perangkat yang telah dirancang. Analisis ini bertujuan untuk mengidentifikasi masalah yang muncul dan mengevaluasi untuk meningkatkan kinerja perangkat.

f. Pembuatan Laporan

Dari seluruh proses yang telah dilalui, disusunlah laporan Tugas Akhir yang merangkum semua hasil dari perangkat yang telah dibuat.

## 1.7 Proyeksi Pengguna

Proyeksi penggunaan ini bertujuan untuk memberikan estimasi target tentang penggunaan produk layanan pengantar makanan di Tel-U Coffee dengan waktu yang berkepanjangan. Selain itu penelitian pengembangan ini ditujukan agar dapat di distribusikan ke masyarakat maupun industri bidang makanan.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Internet of Things (IoT)**

Internet of Things (IoT) merupakan jaringan perangkat yang saling terhubung melalui internet dan dapat berkomunikasi satu sama lain. IoT pada dasarnya melibatkan integrasi sensor, perangkat keras, perangkat lunak, dan koneksi internet untuk memungkinkan objek-objek saling terhubung dan berkomunikasi dalam sebuah infrastruktur digital. Objek-objek ini dapat mengirim serta menerima data, membuat keputusan berdasarkan informasi yang dikumpulkan, dan berinteraksi satu sama lain untuk mencapai tujuan tertentu [3], [4].

##### **2.1.1 Arsitektur IoT**

Arsitektur IoT merupakan desain konsep yang menggambarkan cara perangkat IoT saling terhubung, berkomunikasi, dan bekerja sama untuk mencapai tujuan tertentu. Arsitektur ini terbagi ke dalam empat lapisan utama, yaitu[5]:

###### **1. Lapisan Perangkat**

Lapisan ini meliputi perangkat IoT seperti sensor dan perangkat lain yang terhubung ke internet. Perangkat ini mampu berinteraksi dengan jaringan dan layanan melalui protokol komunikasi yang diatur oleh lapisan jaringan. Kekuatan perangkat terletak pada kemampuannya untuk dilengkapi dengan sensor guna mengumpulkan data, lalu mengirimkannya melalui jaringan. Selain itu, perangkat juga dilengkapi dengan aktuator yang memungkinkan mereka memberikan respons, misalnya menyalakan atau mematikan lampu.

###### **2. Lapisan Jaringan**

Lapisan ini mencakup teknologi dan jaringan yang mendukung perangkat IoT untuk terhubung dengan internet. Beberapa protokol jaringan yang sering digunakan dalam IoT meliputi WiFi, Bluetooth, ZigBee, dan LoRaWAN. Gateway memiliki peran penting dalam menyediakan koneksi antara perangkat dan cloud, sekaligus memungkinkan transmisi perintah dari cloud ke perangkat yang akan

dieksekusi oleh aktuator. Dengan keberadaan gateway, solusi IoT menjadi lebih efektif dan efisien.

### 3. Lapisan Platform

Lapisan ini mencakup perangkat lunak dan layanan untuk mengelola data dari perangkat IoT, termasuk manajemen perangkat, analisis data, dan integrasi cloud. Prosesor data streaming memastikan pengiriman data ke danau data dan aplikasi kontrol secara aman dan efisien tanpa kerusakan.

### 4. Lapisan Aplikasi

Lapisan ini berfokus pada aplikasi yang memproses dan menganalisis data IoT, dengan penerapan di berbagai sektor seperti manufaktur, kesehatan, dan transportasi. Aplikasi ini memungkinkan pengguna memantau, mengontrol, dan mengotomatisasi perangkat pintar melalui platform berbasis web atau mobile.

#### 2.1.2 Kodular

Kodular adalah platform berbasis web yang menyediakan alat serupa dengan MIT App Inventor untuk merancang aplikasi Android menggunakan block programming. Keunggulan Kodular terletak pada fitur komponen palet yang lebih kompleks, dukungan plugin monetisasi bawaan, dan kemudahan penggunaan tanpa perlu menginstal perangkat lunak tambahan. Pengguna hanya memerlukan web browser untuk mengetik parameter pada program blocks tanpa harus menulis kode dari awal[6].

Kodular memungkinkan pembuatan aplikasi Android secara efektif dan efisien dengan metode "drag and drop" pada program blocks yang tersedia. Selain itu, fitur Kodular Companion memungkinkan pengguna untuk melihat dan menguji aplikasi yang sedang dibuat secara langsung tanpa perlu mengekspor aplikasi tersebut. Jika terjadi masalah, blocks dapat segera diperbaiki[6].

## 2.2 Robotika

Robot dirancang untuk berinteraksi dengan lingkungan sekitarnya dan menjalankan tugas-tugas yang sulit, berbahaya, atau repetitif yang biasanya dilakukan oleh manusia. Robot hadir dalam berbagai bentuk dan ukuran, mulai dari robot humanoid yang menyerupai manusia hingga robot industri yang besar dan kompleks. Secara umum, robot terdiri dari beberapa komponen utama, seperti perangkat keras (termasuk sensor, aktuator, dan sistem penggerak), perangkat lunak (yang mengatur perilaku dan tugas yang dijalankan oleh robot), serta sistem pemrosesan data (yang memungkinkan robot untuk menerima, menganalisis, dan merespons informasi dari lingkungannya). Robot digunakan di berbagai sektor, termasuk industri, militer, kesehatan, pertanian, eksplorasi luar angkasa, hingga kebutuhan rumah tangga [4] .

Teknologi robot adalah sebuah alat yang dirancang untuk membantu manusia dalam menjalankan tugas-tugasnya dan memiliki sejumlah keunggulan. Keunggulan tersebut memungkinkan robot digunakan di area yang sulit dijangkau atau berbahaya bagi manusia[7].

### 2.2.1 BELLABOT

BellaBot adalah robot pelayan pintar yang dikembangkan oleh Pudu Robotics, dirancang untuk membantu pengantaran makanan atau barang di restoran, hotel, atau tempat layanan lainnya. BellaBot mampu mengantarkan makanan atau barang secara otomatis, menghindari rintangan, dan mengantarkan barang dengan presisi. Robot ini dirancang

untuk meningkatkan efisiensi operasional dan memberikan pengalaman unik dan menyenangkan bagi pelanggan [8], [9].



Gambar 2. 1 Robot BELLABOT

### 2.2.2 Sistem Pintu Geser Otomatis

Pintu geser otomatis berfungsi tanpa memerlukan tenaga manusia untuk membuka atau menutupnya, karena dilengkapi dengan sensor inframerah yang memungkinkan pintu terbuka dan tertutup secara otomatis. Pintu jenis ini bekerja dengan cara menggeser daun pintu ke arah kiri atau kanan. Untuk mendukung pergerakannya, rel dan roda dipasang di bagian atas dan bawah pintu. Sistem geser pada pintu menjadi pilihan menarik untuk menciptakan kesan ruangan yang lebih luas [10].

### 2.3 Mekanisme Lazy Susan

Lazy Susan, yang juga dikenal sebagai meja putar, adalah perangkat berbentuk lingkaran yang bergerak memutar di atas bantalan. Alat ini biasa diletakkan di meja makan atau dapur untuk memudahkan penyajian makanan. Di samping fungsinya di meja, Lazy Susan juga dipasang di lemari dapur untuk meningkatkan kemudahan akses penyimpanan. Cukup dengan satu putaran, semua barang di atasnya dapat dijangkau dengan mudah. Material pembuatannya beragam, mulai dari kayu, plastik, kaca, marmer, baja anti karat,

hingga kawat. Pemilihan desainnya disesuaikan dengan tipe pintu, posisi lemari, dan kesukaan penggunanya [11].

Mekanisme putar Lazy Susan memerlukan bantalan pada batang yang membuat rak berputar pada sudut tertentu. Beberapa jenis penyelenggara kabinet yang dapat diputar, juga disebut Penyelenggara Kabinet Lazy Susan meliputi[12]:

- Lazy Susan Berbentuk D (180 derajat)
- Lazy Susan Berbentuk Ginjal (270 derajat)
- Lazy Susan Berbentuk Bulat (360 derajat)

## 2.4 Sistem Kontrol

Sistem kontrol atau sistem kendali merupakan gabungan dari sejumlah komponen yang saling berhubungan untuk mencapai tujuan tertentu, yaitu mengatur atau mengendalikan suatu sistem. Sistem ini terbagi menjadi dua jenis utama, yaitu sistem kontrol terbuka (*Open loop Controls*) dan sistem kontrol tertutup (*Close Loop Controls*)[13].

### 2.4.1 Open Loop Control (Sistem Kontrol Terbuka)

Sistem kontrol terbuka atau *Open Loop Control* adalah sistem kendali di mana keluaran (output) tidak memengaruhi proses atau pengontrolan yang dilakukan. Dalam sistem kontrol loop terbuka, sinyal keluaran tidak memiliki umpan balik (feedback) yang digunakan sebagai alat pengukuran (measurement) untuk dibandingkan dengan sinyal masukan[14].

### 2.4.2 Closed Loop (Sistem Kontrol Tertutup)

Sistem kontrol loop tertutup adalah sistem kendali yang dilengkapi dengan umpan balik, di mana sinyal keluaran digunakan untuk memperbaiki sinyal itu sendiri. Sinyal kesalahan yang berasal dari pergerakan akan dikembalikan ke pengontrol sebagai selisih antara sinyal masukan dan sinyal kesalahan dari umpan balik tersebut[14]. Sistem kontrol tertutup dapat dibedakan beberapa jenis, antara lain adalah kontrol *Proportional Integral Derivative* (PID).

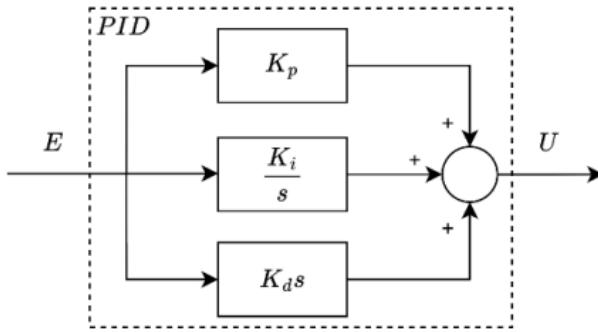
## 2.5 Kontrol PID

Kontrol PID adalah sistem kendali yang berfungsi untuk mengukur tingkat presisi dalam sistem instrumentasi, dengan memanfaatkan umpan balik sebagai fitur utamanya [15]. Kontroler PID dibagi dalam dua bentuk, yaitu kontroler PID analog dan kontroler PID digital. Pada penelitian ini kontroler PID yang digunakan yaitu kontroler PID digital karena pengolahannya diimplementasikan melalui pemrograman dan dijalankan oleh komputer atau mikrokontroler. Sedangkan pada kontrol PID analog pengolahannya memerlukan sejumlah komponen analog untuk merealisasikan persamaan matematis PID.

### 2.5.1 Kontroler PID Digital

Kontroler PID digital merupakan jenis kontroler PID yang diimplementasikan melalui pemrograman dan dijalankan oleh komputer atau mikrokontroler. Dalam proses perhitungannya, variabel *error* dan *last error* dimanfaatkan untuk menentukan aksi kontrol. Pada setiap iterasi program, nilai *error* diperbarui menggunakan data sensor terbaru, sedangkan nilai *error* dari iterasi sebelumnya disimpan dalam variabel *last error* [16]. Pada penelitian ini kontroler PID digital yang dipakai adalah kontroler PID pararel.

Kontroler PID paralel merupakan salah satu jenis kontrol PID yang bekerja dengan tiga jalur terpisah secara bersamaan, yaitu jalur proporsional (P), integral (I), dan turunan (D). Setiap jalur memberikan kontribusi secara mandiri terhadap sinyal kontrol akhir. Struktur ini memungkinkan setiap jalur dapat disetel secara independen, sehingga karakteristik respons sistem dapat diatur dengan lebih presisi. Proses tuning umumnya dilakukan dengan menentukan nilai K<sub>p</sub>, K<sub>i</sub>, dan K<sub>d</sub> secara bertahap, kemudian menggabungkannya dalam persamaan kontrol. Berikut adalah diagram blok kontroler PID.



Gambar 2. 2 Diagram Blok Kontroler PID pararel

Pada gambar 2.2 PID memiliki input  $E$ , output  $U$ , dan parameter PID, masing-masing  $K_p$ ,  $K_i$ , dan  $K_d$ . Setiap komponen P, I, dan D bekerja secara mandiri. Perubahan pada satu koefisien hanya memengaruhi besarnya aksi yang terkait dengan komponen tersebut tanpa memengaruhi komponen lainnya. Struktur paralel ini menjamin bahwa penyesuaian pada satu jalur tidak secara langsung berdampak pada jalur yang lain, sehingga mempermudah proses pengaturan sekaligus menjaga kestabilan sistem[17].

Dalam sistem digital, pemrosesan sinyal kontrol oleh PID digital dilakukan pada waktu-waktu diskrit. Pada proses ini, sinyal analog dikonversi menjadi sinyal digital, kemudian sinyal error diolah, dan hasilnya dikonversi kembali ke sinyal analog. Semua tahapan tersebut berlangsung pada interval waktu *sampling* ( $T_c$ ) tertentu[18]. Adapun nilai-nilai variabel seperti pada persamaan dibawah ini:

$$u[k] = K_p \cdot e[k] + K_i \cdot T \cdot \sum_{i=0}^k e[i] + K_d \cdot \frac{e[k] - e[k-1]}{T} \quad (1)$$

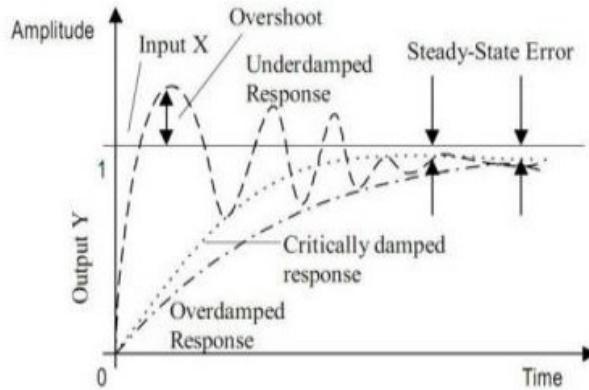
$$\text{Error} = SP \text{ (Setpoint)} - PV \text{ (Present value)} \quad (2)$$

Setpoint (SP) adalah suatu parameter nilai acuan atau nilai yang ingin dicapai. Present Value (PV) adalah nilai pembacaan sensor saat itu atau variabel terukut yang di umpan balikan oleh sensor[16].

#### 2.4.3 Respon Transient

Saat input atau output sistem berubah dengan cepat, mereka membutuhkan waktu untuk menyesuaikan atau merespons perubahan

tersebut. Gambar berikut menunjukkan perilaku sistem selama proses penyesuaian ini. Perilaku ini dikenal sebagai respon transien atau transisi.



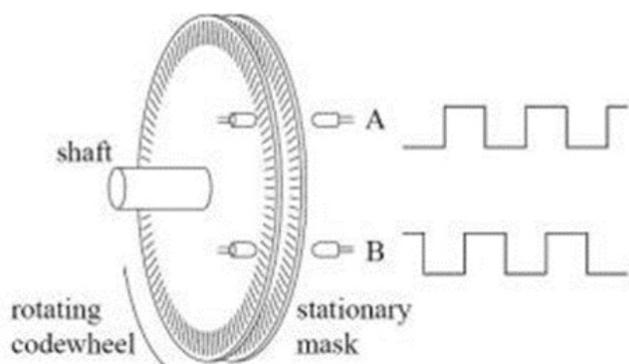
Gambar 2. 3 Bentuk Sinyal Tanggapan Transien

Sistem orde kedua adalah pendekatan yang paling umum digunakan dalam sistem kendali. Sistem ini memiliki beberapa parameter untuk mengukur kualitas respons transien, yaitu sebagai berikut:

1. *Rise Time (TR)*: Waktu yang diperlukan untuk respons untuk mencapai nilai steady-state pertama kali, dihitung dari  $t = 0$  hingga respons memotong sumbu steady-state.
2. *Settling Time (TS)*: Waktu yang menunjukkan bahwa respons telah mencapai 5%, 2%, atau 0,5% dari nilai steady-state
3. *Delay Time (TD)*: Waktu yang menunjukkan keterlambatan respons output terhadap input, dihitung dari  $t = 0$  hingga respons mencapai 50% dari nilai steady-state.
4. *Overshoot (MP)*: Besarnya nilai maksimum dari respons yang melebihi nilai steady-state, ditunjukkan sebagai perbandingan relatif terhadap nilai steady-state.
5. *Offset*: Perbedaan atau kesalahan antara nilai yang diinginkan dan nilai sebenarnya yang dicapai sistem[19].

## 2.6 Rotary Encoder

Rotary encoder adalah perangkat elektromekanis yang berfungsi untuk memantau pergerakan dan posisi suatu objek berputar. Umumnya, rotary encoder memanfaatkan sensor optik untuk menghasilkan pulsa secara berurutan yang kemudian dapat diterjemahkan menjadi informasi gerakan, posisi, dan arah putaran. Dengan demikian, sudut putar poros dapat dikonversi menjadi data digital yang diproses oleh sistem kendali. Rotary encoder banyak digunakan dalam aplikasi seperti pengendalian robot, penggerak motor (motor drive), dan berbagai sistem yang membutuhkan deteksi posisi dengan presisi tinggi [20].



Gambar 2. 4 Susunan Incremental Rotary Encoder

Incremental encoder terdiri dari dua track atau single track dan dua sensor yang disebut channel A dan B. Ketika poros berputar, pada contoh Gambar 2.4, deretan pulsa akan muncul di masing-masing channel pada frekuensi yang proporsional dengan kecepatan putar sedangkan hubungan fasa antara channel A dan B menghasilkan arahputaran,. Dengan menghitung jumlah pulsa yang terjadi terhadap resolusi piringan maka putaran dapat diukur. Untuk mengetahui arah putaran, dengan mengetahui channel mana yang leading terhadap channel satunya dapat kita tentukan arah putaran yang terjadi karena kedua channel tersebut akan selalu berbeda fasa seperempat putaran (quadrature signal). Seringkali terdapat output channel ketiga, disebut INDEX, yang menghasilkan satu pulsa per putaran berguna untuk menghitung jumlah putaran yang terjadi [21].

Rotary encoder incremental sebagai sensor utama untuk mendeteksi posisi pintu. Rotary encoder bekerja dengan menghasilkan pulsa digital setiap kali porosnya berputar. Pulsa ini mewakili perubahan posisi sudut, di mana jumlah pulsa yang diterima oleh mikrokontroler berbanding lurus dengan sudut putaran yang telah ditempuh pintu. Dengan membaca pulsa ini secara real-time, mikrokontroler dapat menghitung posisi aktual pintu, serta kecepatan pergerakannya. Pada sistem pintu otomatis ini, rotary encoder digunakan untuk membaca posisi pintu dari kondisi tertutup ( $0^\circ$ ) hingga terbuka penuh ( $180^\circ$ ). Encoder menghasilkan pulsa sebanyak 13.853 untuk pergerakan pintu dari  $0^\circ$  sampai  $180^\circ$ . Berikut adalah perhitungan untuk mengubah nilai pulsa menjadi sudut:

$$\text{Sudut putaran} = \frac{\text{Jumlah pulsa yang terbaca sensor}}{\text{Jumlah pulsa per putaran penuh (PPR)}} \times 180^\circ \quad (1)$$

## 2.7 Penelitian Terdahulu

Penelitian terdahulu yang dilakukan oleh Rifqi Fachrizi, dkk. (2024) dalam penelitiannya yang berjudul "*Mobilisasi Robot Pengantar Makanan Berbasis Odometry dan QR Detection*". Jenis penelitian adalah deskriptif, dengan menggunakan metode pendekatan kualitatif.

Persamaan penelitian sebelumnya dengan penelitian ini adalah sebagai berikut:

- Objek yang diteliti sama-sama membahas terkait base pada robot pengantar makanan

Sedangkan perbedaan penelitian sebelumnya dengan penelitian ini adalah sebagai berikut:

- Subjek dalam penelitian sebelumnya adalah mobilisasi robot pengantar makanan, sedangkan objek dalam penelitian ini adalah pengembangan pintu otomatis dan base pada robot pengantar makanan.
- Fokus pada penelitian sebelumnya adalah keseluruhan robot terutama pada bagian base robot, sedangkan penelitian ini fokus pada bagian pengembangan pintu dan base berbasis *Internet of Things* (IoT).

Berikut adalah komponen yang dipakai di penelitian sebelumnya:

Nama Komponen	Jenis Komponen
STM32 F411CEU6 -WeAct Blackpill V2.0	Mikrokontroler Base
External Rotary Encoder 600 PPR	Sensor External Rotary Encoder
MPU6050	Sensor IMU
PG45 19.2K 24V 7 PPR encoder	Motor DC
BTS7960	Motor Driver
SMT-Power Aki Elektronik 12V 7Ah	Baterai

## BAB III

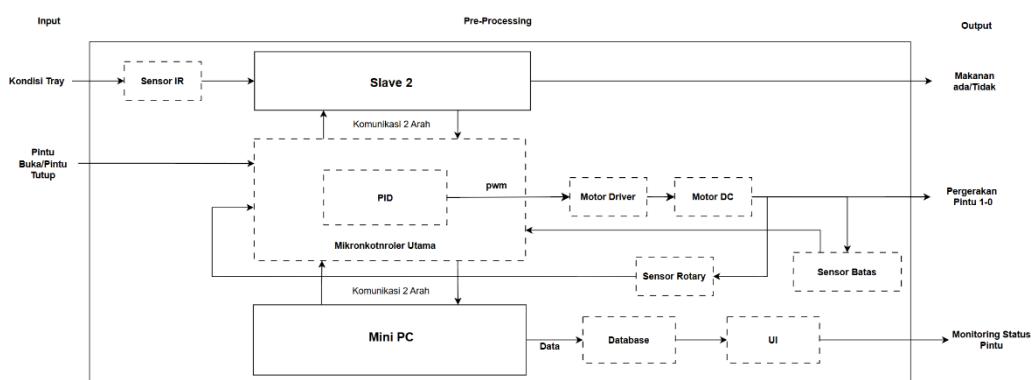
### PERANCANGAN SISTEM

#### 3.1 Desain Sistem

Pada robot ini, desain sistem dirancang untuk menggerakkan pintu geser atau penutup bagian depan pada robot pengantar makanan yang akan terbuka secara otomatis, serta meningkatkan kontrol pergerakan robot. Pada bab ini akan dibahas mengenai sistem kerja, desain perangkat keras dan desain perangkat lunak.

##### 3.1.1 Diagram Blok

Pada penelitian ini terdapat diagram blok sistem sebagai rancangan perangkat keras. Berikut diagram blok sistem dijelaskan pada Gambar 3.1.



Gambar 3. 1 Diagram Blok Sistem pintu Otomatis

Berikut penjelasan mengenai proses input sampai dengan output dari setiap komponen yang digunakan:

##### 1. Input

*Input* dari sistem pergerakan kontrol (*base*) pada robot pengantar makanan berupa koordinat dan mapping, dengan menggunakan sensor Pemetaan. Lalu untuk *input* bagian pintu pada robot yakni, sudut dan nilai keluaran dalam bentuk logika biner (1-0) menggunakan sensor magnetik.

##### 2. Proses

- a. Sensor Rotary dalam sistem pintu berfungsi sebagai pembaca posisi putaran pada sistem pintu dengan memberikan sinyal

berupa pulsa digital yang merepresentasikan jumlah dan arah putaran. Dalam sistem kontrol PID, encoder ini digunakan sebagai umpan balik (feedback) untuk mengetahui seberapa jauh atau seberapa cepat pintu telah bergerak, baik saat membuka maupun menutup.

- b. Mini PC berfungsi menerima data dari sensor pemetaan untuk membuat peta dan menentukan posisi dari robot melalui proses metode SLAM (Simultaneous Localization and Mapping). Data ini diolah melalui algoritma PID Posisi berupa koordinat posisi global. Selanjutnya PID posisi mengirimkan data yang sudah diolah dalam bentuk error koordinat global ( $V_x$ ,  $V_y$ ,  $V\theta$ ) untuk ditransformasikan dan masuk ke rumus kinematik dalam bentuk koordinat lokal.
- c. Mikrokontroler Utama yang digunakan pada sistem ini terdapat proses kontrol PID yang datanya didapat dari sensor *Magnetic* dalam bentuk sudut yang berfungsi untuk mengukur posisi sudut dari pergerakan pintu agar tidak mengalami hentakan. Lalu data tersebut menjadi masukan utama untuk modul kontrol PID yang bertanggung jawab untuk mengatur pergerakan pintu secara akurat dan halus. PID menerima data posisi dari *Magnetic Encoder* dan membandingkannya dengan posisi yang diinginkan untuk menghasilkan sinyal kontrol. Sinyal kontrol dari PID kemudian diteruskan ke Motor Driver, yang mengatur daya untuk Motor DC. Selanjutnya, motor DC menghasilkan gerakan fisik yang menggerakkan pintu robot.
- d. Sensor batas berfungsi sebagai saklar mekanik yang berfungsi untuk mengontrol gerakan dari pintu. Pada saat pintu bergerak menutup, ujung bagian dari pintu akan menyentuh sensor batas yang nantinya akan langsung memberikan sinyal untuk menghentikan pergerakan motor DC guna mencegah pintu bergerak melebihi batas.

- e. Sensor IR berfungsi sebagai pendekteksi adanya makanan atau tidak ada, dimana kondisi tersebut akan dikirim ke esp32 lalu ke raspiberry pi untuk di tampilkan apakah makanan ada atau tidak dan sebagai keamanan makanan jika ada seseorang mencoba untuk mengambil.
  - f. UI berfungsi sebagai platform untuk dapat monitoring dan mengontrol robot dan pintu dengan cara mengambil data base dari *cloud* yakni firebase yang datanya diperoleh dari ESP32 dan Arduino Mega.
3. Output

Keluaran yang dihasilkan dari proses di atas berupa perpindahan robot dengan nilai 1-0, dimana nilai tersebut akan di terima oleh sistem pintu untuk membuka pintu. Pergerakan pintu yang dihasilkan adalah hasil dari seluruh proses kontrol dan penggerak. Sistem memastikan bahwa pintu bergerak secara terkoordinasi sesuai perintah.

### **3.1.2 Fungsi dan Fitur**

Sesuai dengan penjelasan yang terdapat pada latar belakang, sebuah alat akan menjadi solusi untuk masalah tersebut dengan fungsi-fungsi berikut:

1. Pergerakan pintu robot dapat dilakukan secara otomatis baik terbuka maupun tertutup (saat makanan akan diambil).
2. Meningkatkan keakuratan pergerakan pintu saat terbuka maupun tertutup.
3. Robot memiliki kemampuan untuk menghindari objek yang menghalangi dengan mencari jalur yang kosong.

Selain itu, fitur dari sistem yang dirancang:

1. Aplikasi untuk monitoring pintu terbuka/tertutup melalui aplikasi IoT dari jarak jauh.

2. Menjalankan beberapa algoritma dan proses secara bersamaan yang terhubung satu sama lain.
3. Robot dapat bermanuver secara bebas, memiliki keakuratan kecepatan dan posisi.

### 3.2 Desain Perangkat Keras

Sistem yang dirancang menggunakan beberapa komponen perangkat keras mulai dari sensor, mikrokontroler, aktuator, penurun tegangan, *motor driver*, dan baterai.

#### 3.2.1 Desain Pintu



Gambar 3. 2 3D Desain Pintu

Gambar 3. 3 Desain UI Tampilan Layar 2Gambar 3. 4  
3D Desain Pintu

### 3.2.2 Desain UI

Berikut adalah desain antarmuka pengguna (UI) yang dibuat menggunakan Kodular yang terdiri dari dua layar utama, yaitu *Screen1* (Home) dan *Screen2* (Monitoring).

**GO FOR BETTER HABITS  
WITH T.I.F.A!**

TEL-U INTERACTIVE FOOD ASSISTANT



GET STARTED



Gambar 3. 6 Desain UI Tampilan Layar 1

Gambar 3. 7 Diagram Kontroler Pintu Otomatis  
Gambar 3. 8 Desain UI Tampilan Layar 1

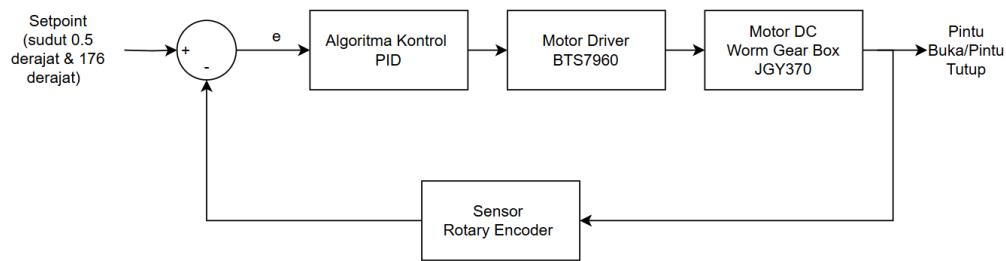


Gambar 3. 5 Desain UI Tampilan Layar 2

Pada gambar 3.3 menampilkan bagian screen1 yang berfungsi sebagai beranda utama yang menampilkan keseluruhan desain dari robot pengantar makanan (TIFA), screen ini juga akan mengantarkan pengguna ke screen2 yaitu monitoring. Lalu pada gambar 3.4 menampilkan screen yang difokuskan pada pemantauan real-time terhadap kondisi sistem. Data yang ditampilkan diambil dari Firebase mencakup:

- Status pintu (terbuka/tertutup)
- Status tray (keluar/masuk)
- Status makanan (ada/tidak ada)

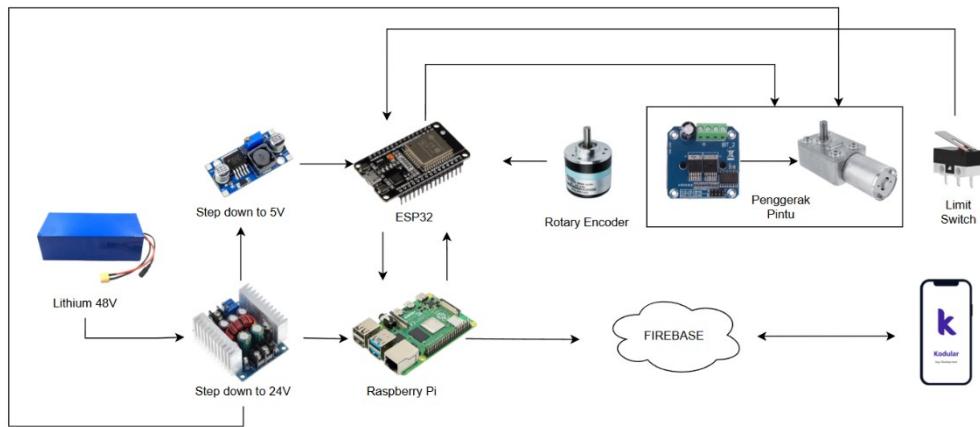
### 3.2.3 Diagram Kontroler Pintu Otomatis



Gambar 3. 9 Diagram Kontroler Pintu Otomatis

Gambar 3.6 merupakan diagram kontroler pintu otomatis dengan menggunakan setpoint berupa sudut dengan nilai 1 derajat (pintu tertutup) dan 176 derajat (pintu terbuka), untuk menentukan posisi target pintu yang ingin dicapai. Mikrokontroler menjalankan algoritma PID yang mengitung error dari selisih antara posisi target (setpoint) dan posisi aktual pintu yang dibaca oleh rotary encoder. Hasil perhitungan PID digunakan untuk mengatur sinyal kendali yang dikirimkan ke motor driver, yang kemudian menggerakkan motor DC worm gear untuk membuka atau menutup pintu. Rotary encoder memberikan umpan balik posisi secara real-time ke mikrokontroler sehingga sistem dapat terus melakukan koreksi hingga pintu mencapai posisi yang diinginkan secara presisi.

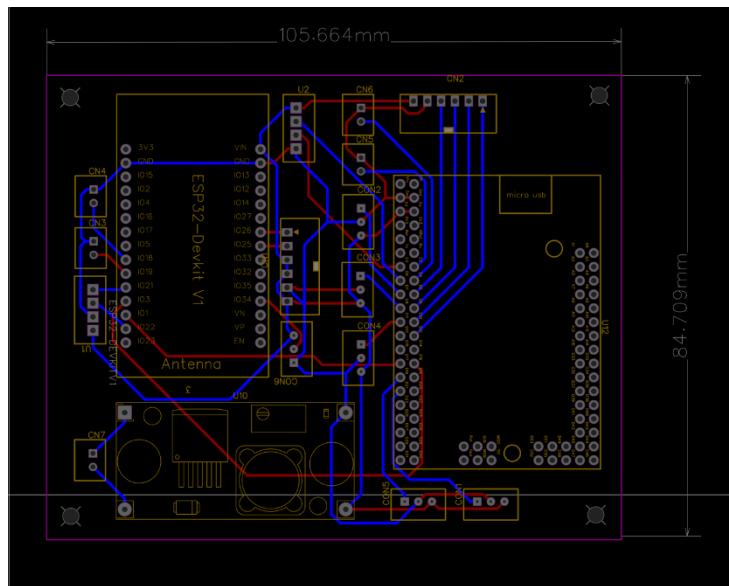
### 3.2.4 Diagram Blok Perangkat Keras



Gambar 3. 10 Diagram Blok Perangkat Keras

Pada gambar 3.7 menampilkan rangkaian dari komponen perangkat keras yang digunakan. Sumber daya utama adalah baterai 3000mAh 48V dan diturunkan ke 24V untuk menyalurkan arus listrik ke motor driver, dan ke mikrokontroler Raspberry Pi dan ESP32, yang sebelumnya telah dihubungkan dengan penurun tegangan menjadi 5V. Motor DC yang digunakan disambungkan ke penurun tegangan 24V.

### 3.2.5 Desain PCB Mikrokontroler



Gambar 3. 13 Desain PCB  
Mikrokontroler

Pada gambar diatas menunjukkan rangkaian PCB berupa mikrokontroler dan sensor yang terdiri dari ESP32, Arduino Mega Pro Mini 2560, sensor IR, sensor ultrasonik, sensor AS5600, Stepdown 5V, serta mengeluarkan pin *input* untuk sensor dan *output* untuk aktuator.

### 3.2.6 Spesifikasi Komponen

#### a. Limit Switch Microswitch KW10-Z1P

Limit switch adalah perangkat elektromekanis yang digunakan untuk mendeteksi pergerakan atau posisi yang dipicu oleh kontak fisik dengan objek. Berfungsi untuk membatasi dan menghentikan gerakan suatu mekanis pada titik tertentu.



Gambar 3. 14 Limit Switch Microswitch KW10-Z1P

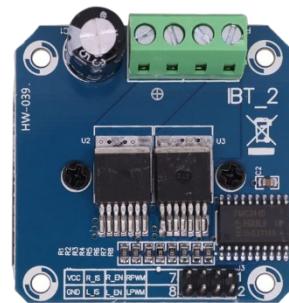
Gambar 3. 15 BTS7960  
Gambar 3. 16 Limit Switch  
Microswitch KW10-Z1P

Tabel 3. 1 Perbandingan Limit Switch

Parameter	Bobot	Skor Penilaian	
		Basic Switch SS-5GL 3PINS SPDT 5A PA97	Microswitch KW10-Z1P
Rating Arus	20	5A	1A
		0.1	0.2
Rating Tegangan	20	125V	125 VAC
		0.2	0.2
Tipe Kontak	30	VCC, NO, NC	NO, NC, COM
		0.3	0.3
Harga	30	8.000	2.000
		0.1	0.2
Total		0.7	0.9

### b. BTS7960

BTS7960 adalah jenis *motor driver* berbasis IC H-Bridge, dirancang untuk mengendalikan kontrol kecepatan dan arah putaran motor DC yang membutuhkan arus yang cukup besar.



Gambar 3. 17 BTS7960

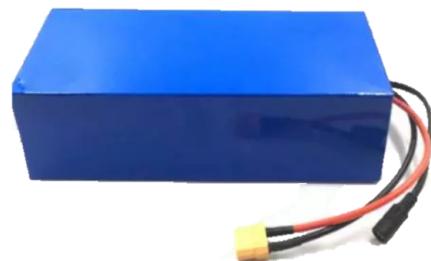
Tabel 3. 2 Perbandingan Motor Driver

Parameter	Bobot	Skor Penilaian
-----------	-------	----------------

		BTS7960	Cytron MDS30C	L298N
Operating Voltage	20	6V – 27V	5V – 25V	5V - 35V
		0.2	0.2	0.2
Continious Current	30	43A	30A	2A
		0.3	0	-0.1
Pinout	10	PWM, EN_RIGHT, EN_LEFT, VCC, GND	PWM, EN, VCC, GND	PWM, DIR1, DIR2
		0.1	0.1	0.1
Harga	10	43.900	460.000	15.000
		0	-0.1	0.1
Total		0.6	0.2	0.3

### c. Baterai Lithium Ion 48V

Baterai Lithium Ion 3000 mAh 48V adalah baterai yang mampu menyimpan energi hingga 1.440 watt/jam, sehingga sangat cocok untuk sistem yang memerlukan pasokan daya stabil dalam durasi yang panjang. Baterai ini sering digunakan dalam sistem yang



Gambar 3. 18 Baterai Lithium Ion 3000 mAh 48V

Gambar 3. 19 Stepdown LM2596  
Gambar 3. 20  
Baterai Lithium Ion 3000 mAh 48V

memerlukan tegangan tinggi dan kapasitas energi besar.

Tabel 3. 3 Spesifikasi Baterai Lithium Ion 3000 mAh 48V

Parameter	Bobot	Skor Penilaian		
		SMT-Power Aki Elektronik 12V 7Ah	Lithium Lifepo4 24V 12Ah	Lithium Ion 3000 mAh 48V
Kapasitas	30	7000 mAh	12000 mAh	3000 mAh
		0.3	0	0.3
Tegangan	25	12 V	24 V	48 V
		0	0.25	0.25
Dimensi	20	64 mm x 94 mm x 151 mm	68 mm x 112 mm x 130 mm	60 mm x 66 mm x 238 mm
		0.2	0.2	0.2
Harga	25	300.000	1.350.000	551.000
		0	-0.25	0
Total		0.4	0.2	0.75

d. Stepdown LM2596 (Penurun tegangan)

Stepdown LM2596 adalah sebuah modul konverter DC yang berfungsi untuk menurunkan tegangan DC yang lebih tinggi menjadi tegangan output stabil 5V.



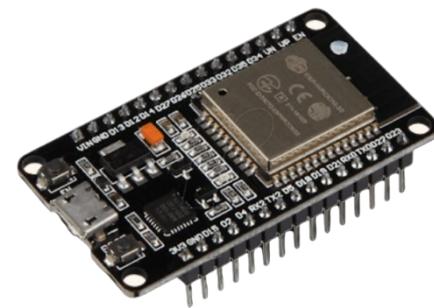
Gambar 3. 21 Stepdown LM2596

Tabel 3. 4 Perbandingan Stepdown

Parameter	Bobot	Skor Penilaian	
		LM2596 (3 A)	XL4015 (5 A)
Tegangan Input	20	4 – 40 V DC	4 – 38 V DC
		0.2	0.2
Tegangan Output	20	1.25 – 37 V	1.25 – 36 V
		0.2	0.2
Proteksi	30	over-current/min imal, perlu heatsink	Over-temp, short-circuit, current limit 8 A
		0	0.2
Harga	30	7.350	15.000
		0.3	0
Total		0.7	0.6

#### e. ESP32 DEV-KIT

ESP32 DEV-KIT adalah mikrokontroler yang dirancang untuk aplikasi berbasis Wifi dan Bluetooth. Memiliki banyak pin *input/output* (GPIO), serta dirancang untuk mengkonsumsi daya yang sangat rendah.



Gambar 3. 22 ESP32 DEV-KIT

Gambar 3. 23 Rotary Encoder  
E38S6G5 600B G24NGambar 3.  
24 ESP32 DEV-KIT

Tabel 3. 5 Perbandingan mikrokontroler utama

Parameter	Bobot (%)	Skor Penilaian	
		ESP32 Dev-Kit	ESP8266 NodeMCU
Mikroprosesor	20	Xtensa Dual-Core 32 Bit LX6	Xtensa Single-core 32-bit L106
		0.2	0.1
Tegangan Operasi	10	3.3V	3.3V
		0.1	0.1
Tegangan Input	10	7-12 V	7-12V
		0.1	0.1
GPIO	30	30 pin GPIO	17 pin GPIO
		0.3	0.1
Harga	20	75.000	52.000
		0.3	0.2
Total		1.00	0.6

f. Rotary Encoder E38S6G5 600B G24N

Rotary Encoder E38S6G5 600B G24N adalah E38S6G5-600B-G24N adalah encoder incremental dua fase dengan output NPN open-collector dan resolusi 600 PPR. Cocok untuk aplikasi industri dan presisi tinggi.



Gambar 3. 25 Rotary Encoder E38S6G5 600B G24N

Tabel 3. 6 Perbandingan Sensor Rotary

Parameter	Bobot	Skor Penilaian	
		E38S6G5 600B G24N	E38S6G5-400B-G24N
Output	30	A/B open-collector NPN, dua kanal	A/B OC NPN
		0.3	0.3
Resolusi	25	600 PPR (2 400 CPR)	400 PPR
		0.25	0
Tegangan Operasi	20	DC 5–24 V	DC 5–24 V
		0.2	0.2

Harga	25	228.500	225.000
		0.25	0.25
Total		1.00	0.75

g. Motor GearBox DC JGY 370 Worm Gear

Motor GearBox DC JGY 370 adalah jenis motor DC yang dilengkapi dengan gearbox berbasis roda cacing (worm gear).



Gambar 3. 26 Motor GearBox DC JGY 370

Gambar 3. 27 Raspberry Pi 4  
Gambar 3. 28  
Motor GearBox DC JGY 370

Tabel 3. 7 Spesifikasi Motor GearBox DC

Parameter	Bobot	Skor Penilaian	
		JGY-370	JGA-25
Tegangan kerja	30	6–12 V	6–12 V
		0.3	0.3
Kecepatan output	25	10–100 RPM	30–500 RPM
		0.25	0
Torsi maksimum	20	±15 kg.cm	±5 kg.cm
		0.2	0.2
Harga	25	93.000	96.000
		0.25	0.25
Total		0.95	0.75

h. Raspberry Pi 4

Raspberry Pi 4 Model B adalah komputer mini dengan prosesor ARM Cortex-A72 quad-core 64-bit yang berjalan pada 1,5 GHz dan kapasitas RAM maksimal 8 GB. Komputer mini ini memiliki fitur konektivitas seperti Wi-Fi, Bluetooth, Gigabit



Ethernet, dan juga mendukung output dual HDMI 4K, USB 3.0, dan 40 pin GPIO. Ini membuatnya ideal untuk berbagai aplikasi seperti komputasi *edge*, Internet of Things (IoT), otomasi, server skala kecil, dan sistem robotika berbasis awan. Raspberry Pi 4 memungkinkan pengembangan sistem tertanam dan aplikasi jaringan yang sangat fleksibel berkat kompatibilitasnya dengan sistem operasi berbasis Linux, seperti Raspberry Pi OS dan Ubuntu.

Tabel 3. 8 Spesifikasi Raspberry Pi 4

Gambar 3. 29 Raspberry Pi 4

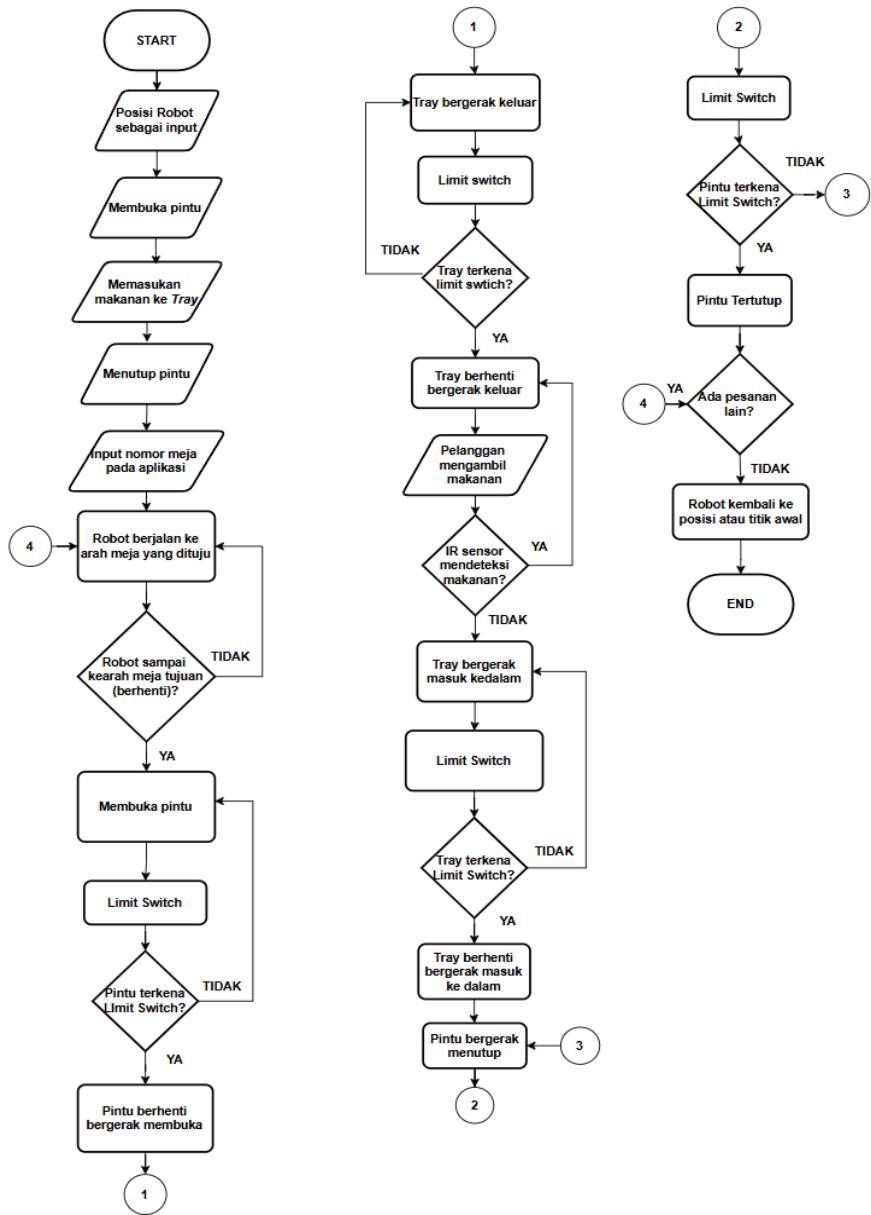
Parameter	Bobot	Skor Penilaian	
		Raspberry Pi 4 Model B	Raspberry Pi 3 Model B+
RAM	30	2 GB / 4 GB / 8 GB LPDDR4	1 GB LPDDR2

		0.3	0.3
Kecepatan output	25	10–100 RPM	30–500 RPM
		0	0
Port USB	25	$2 \times$ USB 3.0, $2 \times$ USB 2.0	$4 \times$ USB 2.0
		0.25	0
GPIO	20	40 pin	40 pin
		0.20	0.20
Total		0.75	0.5

### 3.3 Desain Perangkat Lunak

Dalam sistem ini digunakan sebuah algoritma sehingga sistem bekerja sesuai dengan urutan yang diinginkan. Berikut merupakan *flowchart* perancangan perangkat lunak keseluruhan. *Flowchart* tersebut menjelaskan bagaimana proses *input* sampai *output*.

### 3.3.1 Spesifikasi Sub Sistem Pintu dan *Tray*



Pada gambar 3.16 Proses dimulai dengan robot berada di posisi awal dan memulai pengantaran dengan membuka pintu untuk memasukkan makanan ke dalam *tray*. Setelah makanan masuk ke dalam *tray*, pintu robot ditutup dan aplikasi digunakan untuk menentukan arah pengantaran untuk memasukkan nomor meja tujuan. Selanjutnya, robot bergerak menuju meja pelanggan yang

telah ditentukan. Robot berhenti di meja dan secara otomatis membuka pintu. Limit switch mengontrol apakah pintu telah terbuka sepenuhnya.

Setelah itu, *tray* bergerak keluar hingga menyentuh limit switch untuk memastikan posisinya benar. Pelanggan kemudian mengambil makanan, yang terdeteksi oleh sensor IR. Setelah makanan diambil, *tray* bergerak kembali masuk ke dalam hingga menyentuh limit switch untuk memastikan tray berada di posisi aman. Setelah *tray* tertutup sepenuhnya, pintu robot akan kembali menutup secara otomatis. Robot akan melanjutkan pengantaran ke meja berikutnya jika ada pesanan tambahan. Namun, jika tidak ada pesanan baru, robot akan kembali ke posisi awal atau ke titik awal untuk bersiap untuk tugas berikutnya. Proses ini menjamin pengantaran makanan yang efektif dan otomatis melalui penggunaan sensor dan sistem kontrol.

## **BAB IV**

### **HASIL DAN ANALISIS**

#### **4.1 Hasil Percobaan**

Pada bagian ini, aktuator dan sensor akan diuji untuk mendapatkan respon sistem terhadap kondisi yang sesuai dan serta menyajikan data yang diperoleh dari hasil percobaan. Data dikumpulkan melalui berbagai uji coba yang menguji kinerja masing-masing komponen dan kinerja sistem secara keseluruhan. Data yang disajikan berupa bentuk tabel atau narasi untuk memudahkan pemahaman.

##### **4.1.1 Pengujian Motor DC JGY370 dan Driver BTS7960**

Pengujian dilakukan dengan mengukur kinerja motor DC yang di kontrol oleh driver BTS7960. Data yang diambil merupakan nilai tegangan dan arus dengan menggunakan nilai PWM berbeda yang diberikan pada driver dan kemudian diukur. Berikut ini merupakan langkah-langkah pengujian/kalibrasi Motor DC dan Driver BTS7960.

1. Sambungkan motor DC ke *power supply* 12 VDC melalui motor driver.
2. Jika semua komponen yang diukur sudah tersambung, gunakan multimeter untuk mengukur tegangan dan arus.
3. Lalu atur multimeter ke mode DC Current (A) untuk mengukur arus.
4. Selanjutnya putuskan jalur kabel positif (+) motor yang disambungkan ke motor driver, lalu hubungkan kabel probe merah multimeter ke output motor yang berada di motor driver, sedangkan untuk kabel probe hitam multimeter dihubungkan ke kabel positif (+) motor yang sebelumnya dilepas.
5. Kemudian nyalakan motor dan jalankan sistem, kemudian catat nilai arus yang terbaca pada multimeter, ulangi pengujian sebanyak 10x.
6. Sedangkan untuk mengukur arus atur multimeter ke mode DC Voltage (V). Lalu hubungkan kabel probe merah multimeter ke salah satu output atau terminal pada motor driver, kemudian hubungkan kabel probe hitam multimeter ke output lainnya. (Dalam pengujian ini menggunakan out 1 dan out 2). Cukup tempelkan saja kabel probe tanpa harus memutus kabel yang terhubung.
7. Kemudian nyalakan motor dan jalankan sistem, kemudian catat nilai tegangan yang terbaca pada multimeter, ulangi pengujian sebanyak 10x.

Tabel 4. 1 Pengujian Motor DC dan Driver BTS7960

No.	Nilai PWM	Tegangan Motor (V)									
		P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
1.	255	10.8	10.6	10.6	10.6	10.6	10.7	10.6	10.7	10.7	10.8
2.	191	9.5	9.4	9.5	9.4	9.5	9.4	9.5	9.4	9.5	9.5
3.	128	8.1	8.2	8.2	8.1	8.2	8.2	8.1	8.2	8.1	8.2
4.	64	4.4	4.4	4.3	4.3	4.4	4.4	4.4	4.3	4.3	4.4
5.	0	0	0	0	0	0	0	0	0	0	0

Tabel 4. 2 Pengujian nilai arus motor DC jGY370

Pengujian ke-	Tanpa Beban (A)	Dengan Beban (A)
1	0.12	0.54
2	0.13	0.56
3	0.12	0.58
4	0.13	0.60
5	0.12	0.58
6	0.13	0.56
7	0.12	0.56
8	0.13	0.58
9	0.12	0.60
10	0.13	0.61

Pada pengujian diatas terlihat nilai arus yang di hasilkan tanpa beban stabil dengan rata-rata 0.12A sedangkan pada kondisi dengan beban arus terlihat naik dan turun namun tidak signifikan dan masih dalam batas kategori yang normal dengan nilai rata-rata 0.58A

Berdasarkan pada Tabel 4.1 diatas ditunjukkan bahwa peningkatan nilai PWM sebanding dengan peningkatan kecepatan motor. Driver L298N memiliki kemampuan untuk mengontrol motor DC dengan baik, Nilai PWM rendah menunjukkan hubungan yang hampir linear antara kecepatan motor dan nilai PWM,

dengan sedikit ketidaklinearan. Ini karena gesekan statis yang harus diatasi sebelum motor dapat berputar.

#### 4.1.2 Pengujian Limit Switch

Pengujian ini dilakukan untuk memastikan bahwa mekanisme pendeteksian pada ujung *track* berfungsi dengan baik. Pengambilan data dilakukan dengan 2 *limit switch*. Dilakukan sebanyak 10 kali percobaan dengan menekan tuas pada sensor *limit switch* dengan menggunakan motor DC dan *serial monitor*. Pengujian dilakukan dengan menekan tuas pada sensor *limit switch*, lalu periksa apakah pada *serial monitor* tampil nilai "1" atau "0" dan kondisi dari motor DC yakni berhenti dan berjalan.

Tabel 4. 3 Pengujian *Limit Switch* KW10-ZIP

Pengujian ke-	Status limit switch (high=1/low=0)	Kondisi Motor DC
1	1	Berjalan
2	0	Berhenti
3	1	Berjalan
4	0	Berhenti
5	1	Berjalan
6	0	Berhenti
7	1	Berjalan
8	0	Berhenti
9	1	Berjalan
10	0	Berhenti

Dari tabel 4.3 diatas menunjukkan bahwa percobaan yang telah dilakukan diperoleh hasil yang sesuai, artinya limit switch dapat bekerja dengan baik sesuai dengan keadaan 1 (tuas ditekan/high) yaitu motor berjalan dan ketika keadaan 0 (tuas tidak ditekan/low) motor berhenti.

### 4.1.3 Pengujian Rotary Encoder

Pengujian ini dilakukan menggunakan rotary encoder incremental sebagai sensor utama untuk mendeteksi posisi pintu. Rotary encoder bekerja dengan menghasilkan pulsa digital setiap kali porosnya berputar. Pulsa ini mewakili perubahan posisi sudut, di mana jumlah pulsa yang diterima oleh mikrokontroler berbanding lurus dengan sudut putaran yang telah ditempuh pintu. Dengan membaca pulsa ini secara real-time, mikrokontroler dapat menghitung posisi aktual pintu, serta kecepatan pergerakannya. Pada sistem pintu otomatis ini, rotary encoder digunakan untuk membaca posisi pintu dari kondisi tertutup ( $0^\circ$ ) hingga terbuka penuh ( $180^\circ$ ). Encoder menghasilkan pulsa sebanyak 13.853 untuk pergerakan pintu dari  $0^\circ$  sampai  $180^\circ$ . Berikut adalah perhitungan untuk mengubah nilai pulsa menjadi sudut:

$$\text{Sudut putaran} = \frac{\text{Jumlah pulsa yang terbaca sensor}}{\text{Jumlah pulsa per putaran penuh (PPR)}} \times 180^\circ \quad (1)$$

Proses dimulai ketika motor DC worm gear menggerakkan pintu, dan rotary encoder yang terhubung pada poros motor menghasilkan sinyal pulsa digital. Sinyal tersebut kemudian masuk ke pin interrupt mikrokontroler ESP32, yang akan menghitung jumlah pulsa dan arah putaran. Data ini digunakan untuk menentukan berapa besar error antara posisi aktual pintu dan setpoint yang telah ditentukan dalam algoritma kontrol PID.

Selain itu, data dari rotary encoder sangat penting untuk mengetahui pergerakan pintu karena memberikan feedback langsung yang bersifat kontinu dan presisi. Mikrokontroler memanfaatkan data ini untuk melakukan penyesuaian sinyal PWM ke motor driver agar pintu bergerak halus menuju posisi target. Tanpa encoder, sistem hanya dapat mengandalkan estimasi waktu atau kecepatan motor, yang kurang akurat dan rentan terhadap kesalahan posisi. Dengan demikian, penggunaan rotary encoder meningkatkan presisi, kestabilan, dan keandalan dari seluruh sistem pintu otomatis.

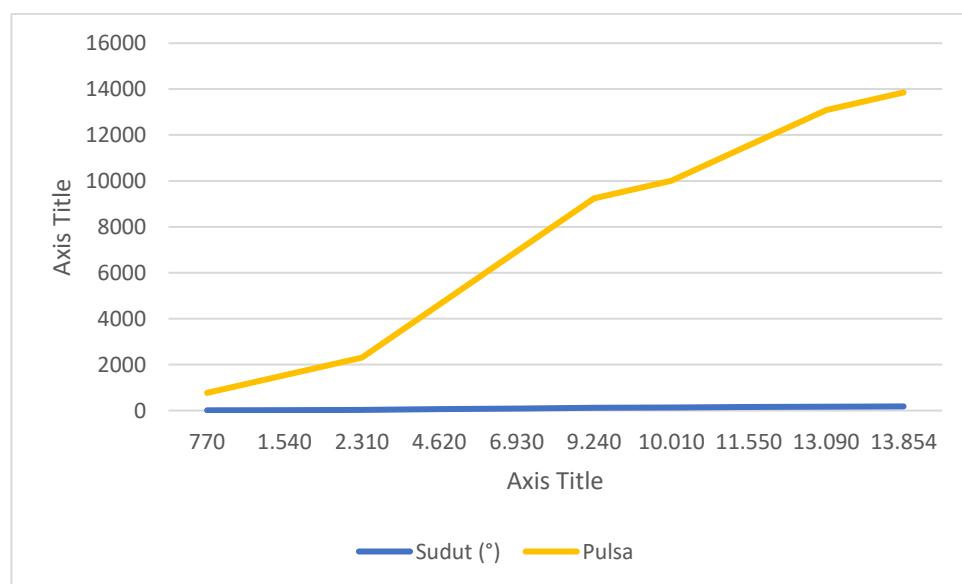
Pengambilan data dilakukan sebanyak 10 kali percobaan. Berikut langkah-langkah pengujian rotary encoder E38S6G5-600B-G24N.

1. Hubungkan channel A, channel B, dan pin VCC, GND rotary encoder pada mikrokontroler ESP32.

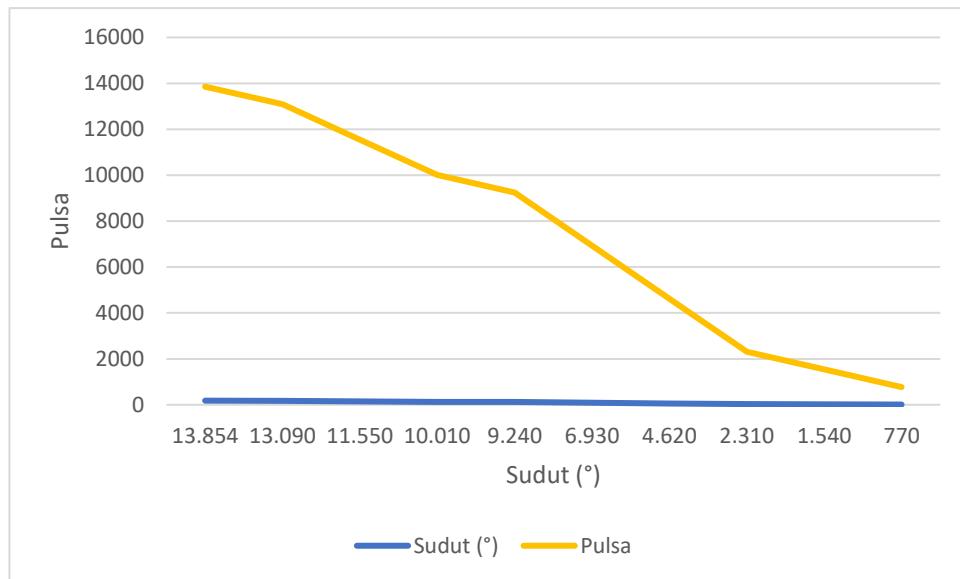
2. Apabila sensor sudah terhubung dengan mikrokontroler, upload program dan buka serial monitor.
3. Putar sensor dari posisi saat pintu tertutup penuh sampai pintu terbuka penuh, lalu catat nilai pulsa yang dihasilkan.

Tabel 4. 4 Pengujian Sensor rotary encoder

No.	Hasil Kalibrasi		
	Sudut (°)	Pulsa yang dihasilkan	Arah
1	10	770	CW
2	20	1.540	CW
3	30	2.310	CW
4	60	4.620	CW
5	90	6.930	CW
6	120	9.240	CW
7	130	10.010	CW
8	150	11.550	CW
9	170	13.090	CW
10	180	13.854	CW



Gambar 4. 1 Grafik saat Rotary Encoder Searah Jarum Jam



Gambar 4. 2 Grafik saat Rotary Encoder Lawan Arah Jarum Jam

Berdasarkan tabel dan grafik hasil kalibrasi sensor rotary encoder di atas, terlihat adanya hubungan yang linier antara sudut putar poros (dalam derajat) dengan jumlah pulsa yang dihasilkan oleh sensor. Setiap kenaikan sudut menghasilkan kenaikan jumlah pulsa secara proporsional, yang menunjukkan bahwa sensor bekerja dengan konsisten dalam mengubah pergerakan mekanis menjadi sinyal digital. Misalnya, pada sudut  $10^{\circ}$  dihasilkan 770 pulsa, dan pada  $90^{\circ}$  dihasilkan 6.930 pulsa, sedangkan pada sudut maksimum  $180^{\circ}$  tercatat 13.854 pulsa. Grafik memperlihatkan garis tren yang mendekati garis lurus, sehingga dapat disimpulkan bahwa akurasi konversi sudut ke pulsa cukup tinggi dan kesalahan pembacaan relatif kecil. Pola ini juga memudahkan proses perhitungan sudut secara *real-time* pada aplikasi kontrol posisi, karena cukup menggunakan rumus konversi linier antara pulsa dan sudut. Hal ini membuktikan bahwa rotary encoder yang digunakan memiliki presisi yang baik dan layak digunakan untuk sistem kontrol otomatis seperti pintu atau robot pengantar makanan.

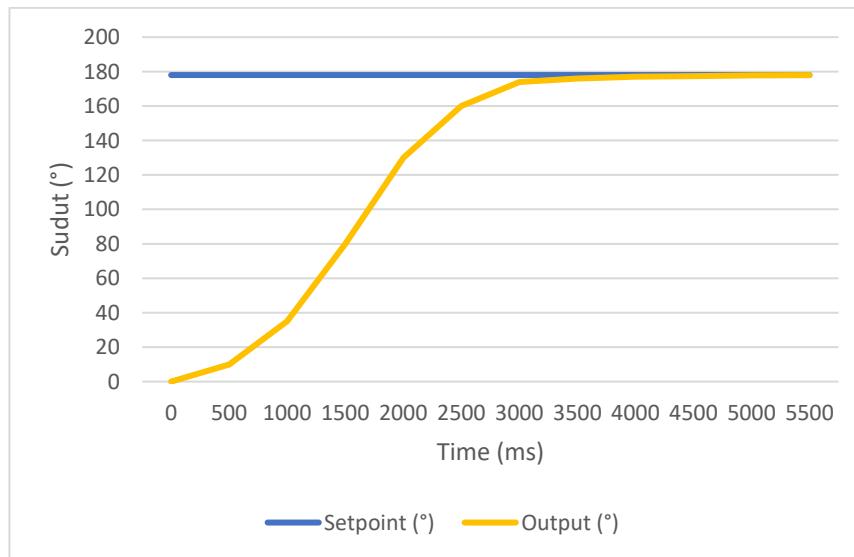
#### 4.1.4 Pengujian Kontrol PID Pintu

Pengujian ini dilakukan untuk mengetahui sistem kontrol posisi pintu otomatis dengan memakai berbagai parameter PID (Proportional, Integral,

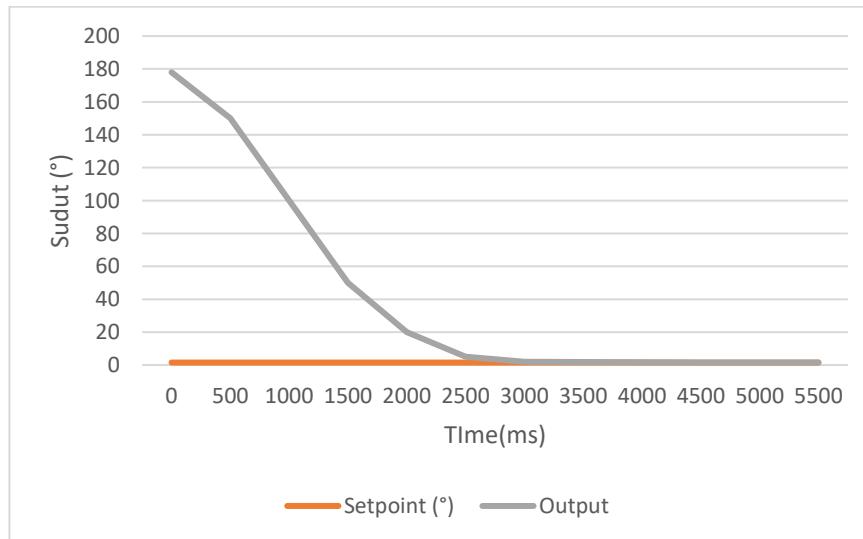
Derivative). Pengujian dilakukan melalui beberapa kali proses membuka dan menutup pintu, di mana setpoint yang digunakan adalah 178 derajat untuk kondisi pintu terbuka penuh dan 1,5 derajat untuk kondisi pintu tertutup. Penggunaan kontrol PID bertujuan untuk mengatur kecepatan serta posisi pintu agar dapat bergerak dengan halus, tanpa overshoot berlebih maupun getaran yang dapat mengganggu kestabilan sistem atau merusak komponen mekanis. Dengan umpan balik dari rotary encoder sebagai sensor posisi, sistem PID mampu meminimalkan error posisi akhir (steady-state error), sehingga pintu dapat berhenti tepat di batas yang diinginkan sesuai setpoint. Pengujian ini juga bertujuan untuk menentukan kombinasi parameter PID ( $K_p$ ,  $K_i$ ,  $K_d$ ) yang paling sesuai dengan karakteristik mekanik pintu dan kebutuhan operasional robot. Tahapan pengujian dimulai dari pengaturan nilai  $K_p$ ,  $K_i$ , dan  $K_d$  dari nol, kemudian dilakukan penyesuaian secara bertahap sambil memantau respon sistem pada Serial Plotter. Nilai  $K_p$  ditingkatkan hingga pintu mampu mencapai setpoint, kemudian nilai  $K_i$  ditambahkan untuk menghilangkan error sisa, dan nilai  $K_d$  digunakan untuk memperhalus gerakan pintu serta mempercepat stabilisasi. Proses ini diulang hingga diperoleh parameter PID yang memberikan pergerakan pintu cepat, akurat, dan stabil menuju setpoint yang ditentukan. Berikut Tahap-tahap pengujian PID:

1. Mengatur nilai ( $K_p$ ,  $K_i$ ,  $K_d$ ) mulai dari nilai 0.
2. Atur pintu otomatis dari posisi tertutup, lalu jalankan dari posisi tertutup ke posisi terbuka.
3. Amati grafik respon diplot menggunakan Serial Plotter.
4. Tingkatkan nilai  $K_p$  sedikit demi sedikit sambil memantau grafik respon pintu.
5. Setelah pintu mampu mencapai setpoint, tambahkan nilai  $K_i$  secara bertahap agar pintu benar-benar berhenti di posisi target tanpa menyisakan error akhir.
6. Jika respon pintu terlihat berosilasi atau terjadi overshoot, tambahkan nilai  $K_d$  untuk memperhalus gerakan pintu dan mempercepat stabilisasi.
7. Periksa apakah pintu bergerak cepat, akurat, dan stabil mencapai setpoint. Jika belum sesuai, ulangi penyesuaian  $K_p$ ,  $K_i$ , dan  $K_d$  hingga didapatkan hasil optimal.

Berikut merupakan hasil pengujian dengan menggunakan nilai ( $K_p, K_i, K_d$ ) yang berbeda-beda:



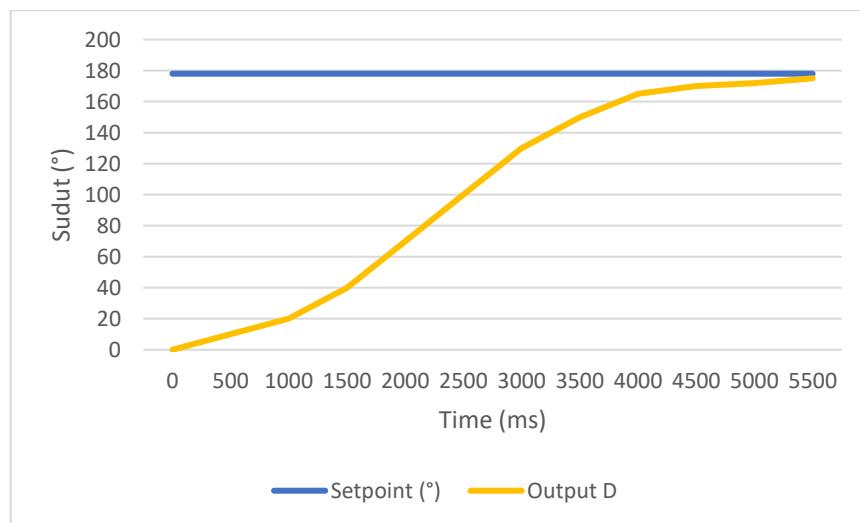
Gambar 4. 3 Grafik saat Pintu Buka



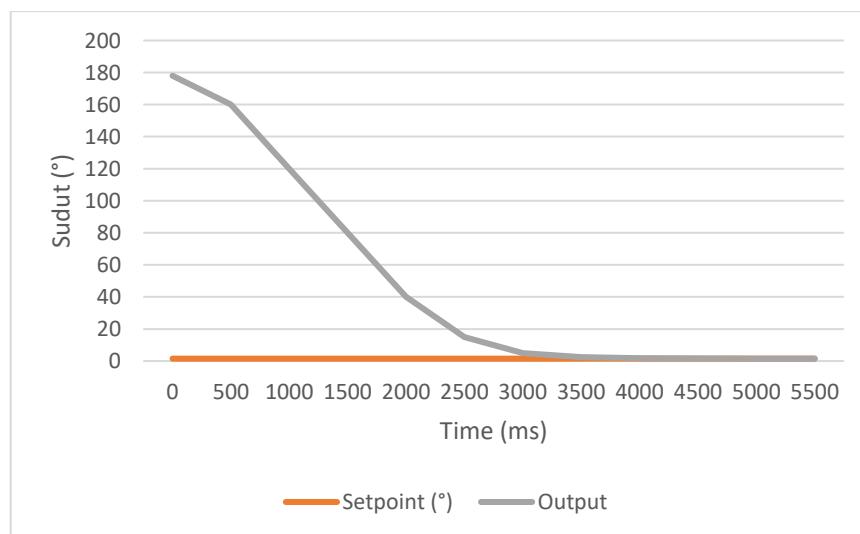
Gambar 4. 4 Grafik saat Pintu Tutup

Berdasarkan grafik PID dengan parameter  $K_p=3,0$ ,  $K_i=0,2$ , dan  $K_d=4,0$ , sistem pintu otomatis menunjukkan karakteristik Penurunan nilai  $K_p$  mengurangi agresivitas respon awal, sehingga laju kenaikan menuju setpoint menjadi lebih lambat dibandingkan konfigurasi ideal. Meskipun demikian,  $K_i$  yang kecil tetap

menjaga penghapusan error akhir, dan  $Kd$  yang besar mempertahankan redaman sehingga sistem tetap stabil tanpa overshoot yang signifikan. Konfigurasi ini sesuai untuk kebutuhan dengan prioritas gerakan yang lebih konservatif, namun mengorbankan sedikit kecepatan respon.



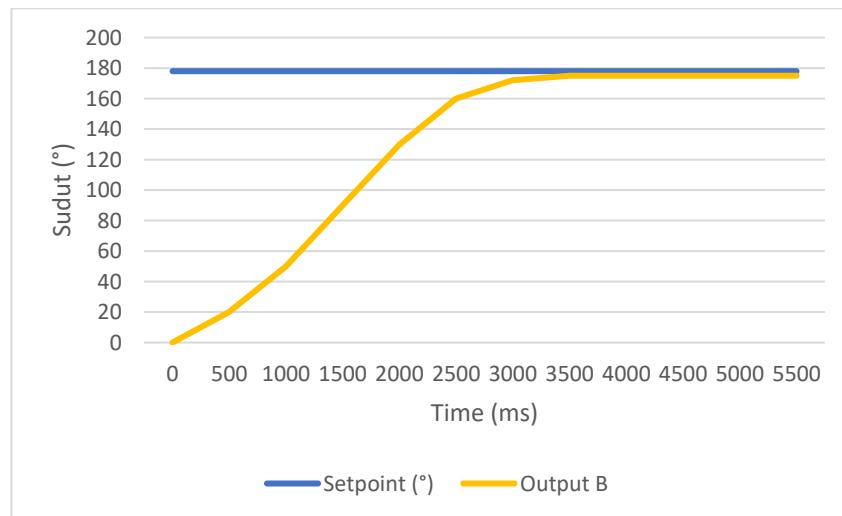
Gambar 4. 5 Grafik saat Pintu Buka (2)



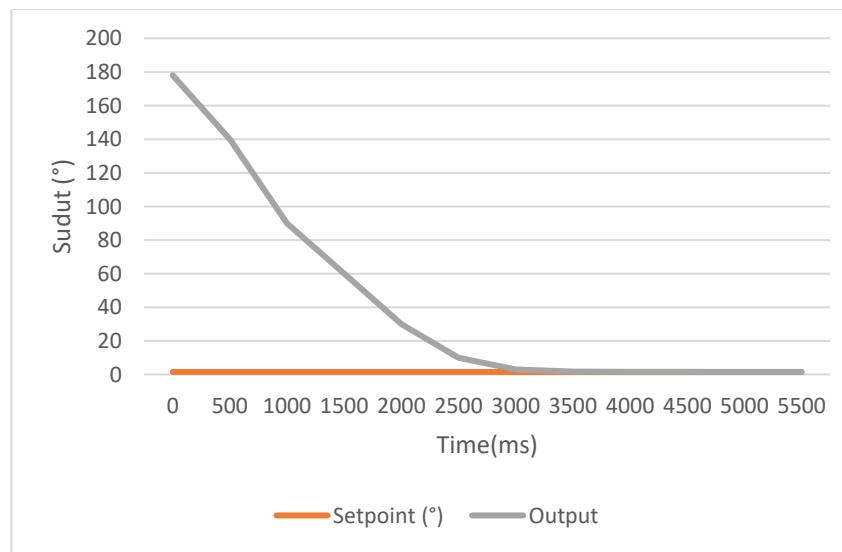
Gambar 4. 6 Grafik saat Pintu Tutup (2)

Berdasarkan grafik PID dengan parameter  $Kp=3,0$ ,  $Ki=0,05$ , dan  $Kd=1,0$ , Dengan nilai  $Kd$  yang lebih rendah, efek redaman menjadi berkurang sehingga sistem cenderung menunjukkan sedikit riak (*underdamped*) saat mendekati

setpoint. Kecepatan respon masih cukup tinggi berkat  $K_p$  yang moderat, dan  $K_i$  kecil tetap efektif menghapus error sisa. Secara keseluruhan, konfigurasi ini memberikan respon yang sedikit lebih gesit dibanding set ideal, namun transisi menjadi kurang halus.



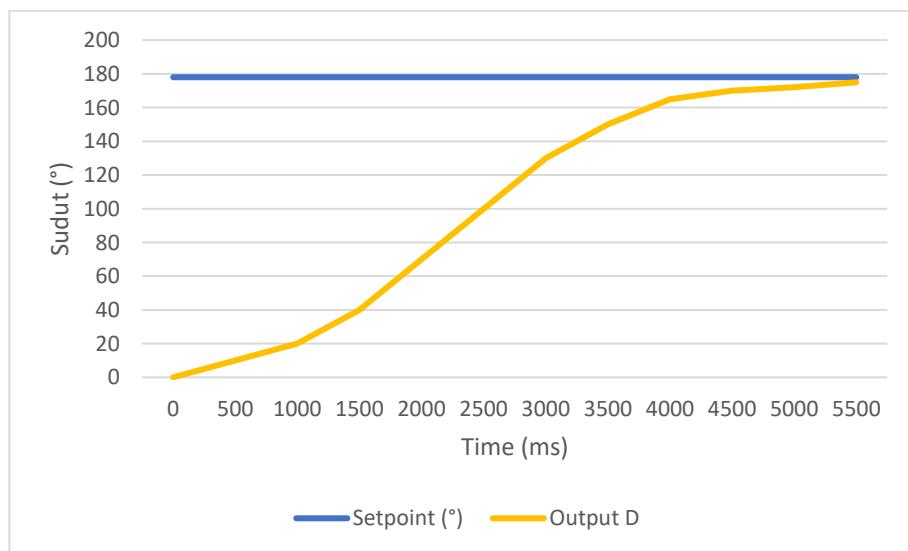
Gambar 4. 7 Grafik saat Pintu Buka (3)



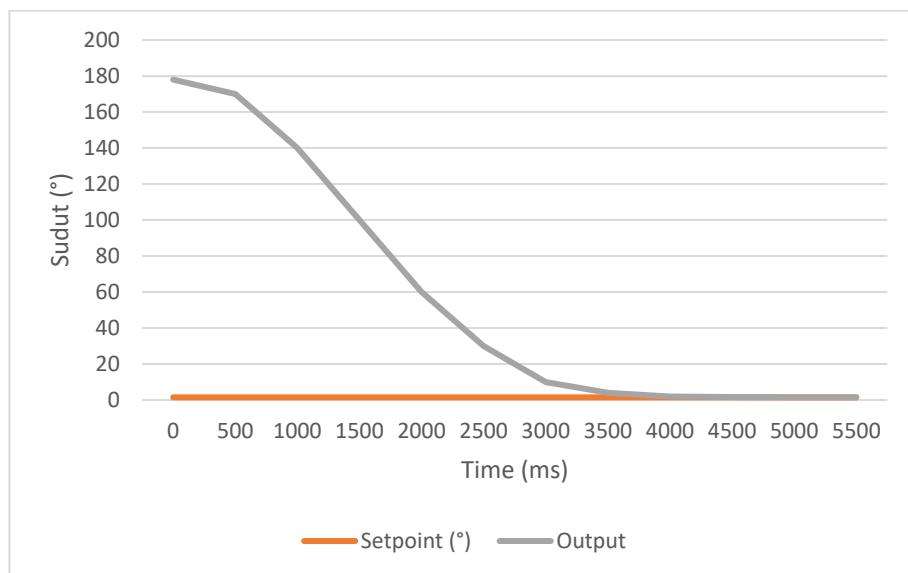
Gambar 4. 8 Grafik saat Pintu Tutup (3)

Berdasarkan grafik PID dengan parameter  $K_p=1.5$ ,  $K_i=0.02$ ,  $K_d=1.0$ , Dengan nilai  $K_d$  yang lebih rendah, efek redaman menjadi berkurang sehingga

sistem cenderung menunjukkan sedikit riak (underdamped) saat mendekati setpoint. Kecepatan respon masih cukup tinggi berkat  $K_p$  yang moderat, dan  $K_i$  kecil tetapi efektif menghapus error sisa. Secara keseluruhan, konfigurasi ini memberikan respon yang sedikit lebih gesit dibanding set ideal, namun transisi menjadi kurang halus.



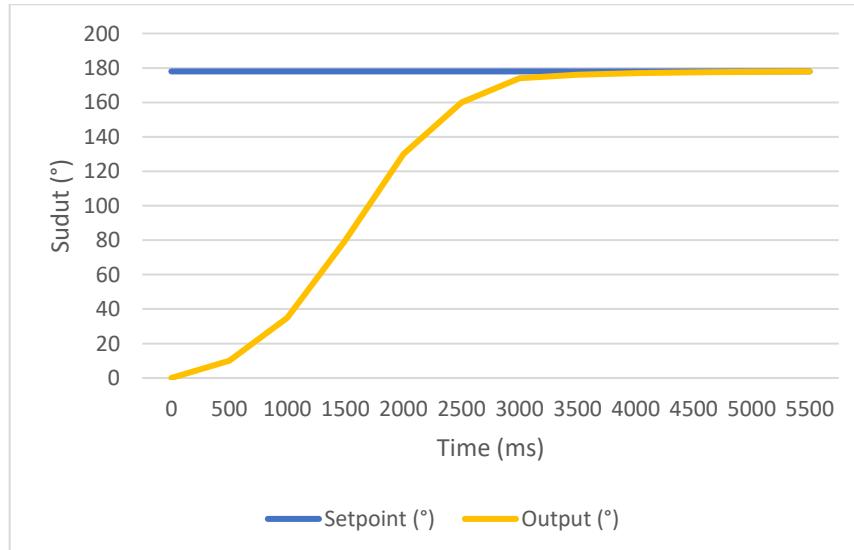
Gambar 4. 9 Grafik saat Pintu Buka (4)



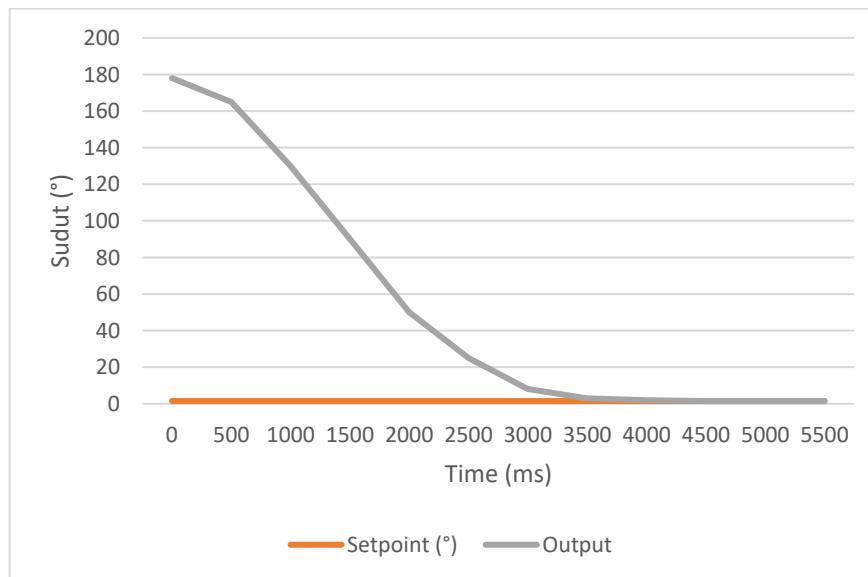
Gambar 4. 10 Grafik saat Pintu Tutup (4)

Berdasarkan grafik PID dengan parameter  $K_p=2.0$ ,  $K_i=0.05$ ,  $K_d=4.0$ . Penurunan nilai  $K_p$  mengurangi agresivitas respon awal, sehingga laju kenaikan menuju setpoint menjadi lebih lambat dibandingkan konfigurasi ideal. Meskipun demikian,  $K_i$  yang kecil tetap menjaga penghapusan error akhir, dan  $K_d$  yang besar

mempertahankan redaman sehingga sistem tetap stabil tanpa overshoot yang signifikan. Konfigurasi ini sesuai untuk kebutuhan dengan prioritas gerakan yang lebih konservatif, namun mengorbankan sedikit kecepatan respon.



Gambar 4. 11 Grafik saat Pintu Buka (5)



Gambar 4. 12 Grafik saat Pintu Tutup (5)

Berdasarkan grafik PID dengan parameter  $K_p=3,0$ ,  $K_i=0,05$ , dan  $K_d=4,0$ , terlihat bahwa sistem pintu otomatis menunjukkan respons yang halus dan stabil. Garis biru sebagai setpoint menggambarkan target sudut pintu yang berubah secara instan dari kondisi tertutup menuju terbuka penuh, lalu kembali tertutup. Garis

merah yang mewakili posisi aktual pintu bergerak mengikuti setpoint dengan baik tanpa menunjukkan overshoot yang signifikan, menandakan bahwa kendali PID mampu mengatur pergerakan secara presisi, Sementara itu, garis hijau sebagai output PID memperlihatkan nilai tinggi di awal untuk memberikan akselerasi, kemudian menurun saat mendekati setpoint dan beralih ke nilai negatif ketika pintu masuk fase penutupan, Hal ini menunjukkan bahwa parameter PID yang digunakan berhasil meminimalkan error steady-state serta menjaga kestabilan pergerakan tanpa osilasi yang berlebihan, Secara keseluruhan, kombinasi parameter ini dapat dikatakan optimal untuk mengendalikan pintu otomatis dengan pergerakan yang presisi, responsif, dan halus,

#### 4.1.5 Pengujian Sistem Jarak Jauh (Internet of Things)

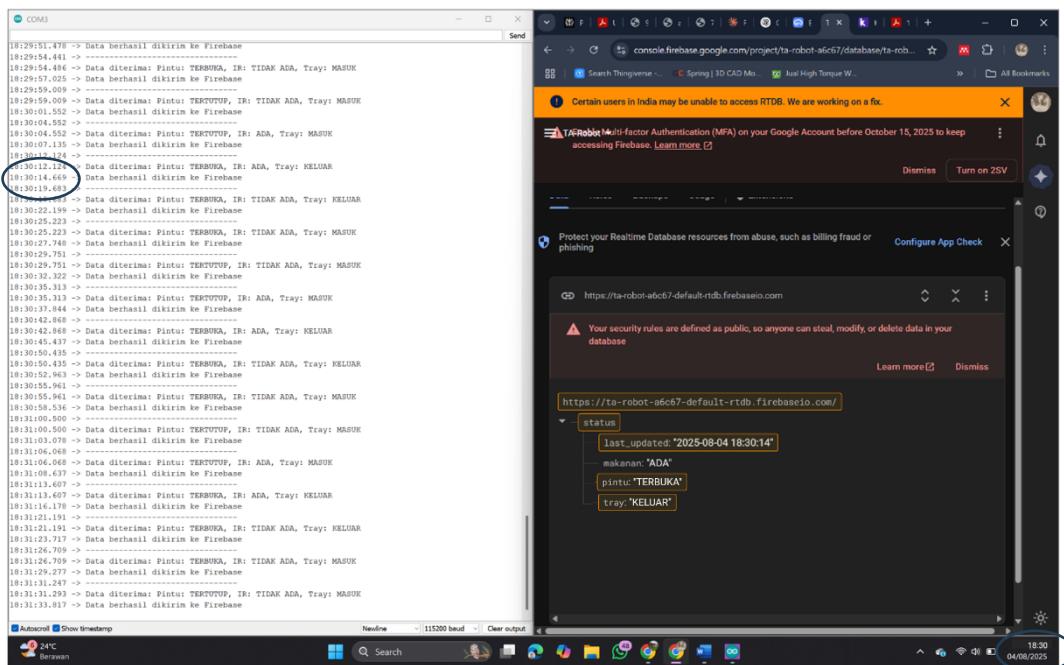
Pengujian ini bertujuan untuk mengevaluasi kinerja sistem pemantauan jarak jauh berbasis *Internet of Things* (IoT) yang telah dirancang, Fokus utama pengujian adalah mengukur *delay* atau jeda waktu antara pengiriman data melalui ESP32 dan penerimaan serta pencatatannya di Firebase melalui jaringan internet, Nilai *delay* diperoleh dari perbedaan waktu antara saat data dicatat di serial monitor sebagai pengirim (transmitter) dan saat data tersebut tersimpan di database Firebase sebagai penerima (receiver),

Tabel berikut menyajikan hasil pengujian delay sistem saat pintu terbuka maupun tertutup, serta menampilkan waktu pencatatan masing-masing sebagai acuan perhitungan delay,

Tabel 4. 5 Pengujian Delay Sistem

No	Status Pintu	Status IR	Status Tray	Waktu di serial monitor	Waktu di database	Delay (s)
1	Tertutup	Ada	Masuk	18:29:59	18:30:01	1
2	Tertutup	Ada	Masuk	18:30:04	18:30:07	3
3	Terbuka	Ada	Keluar	18:30:12	18:30:14	2
4	Terbuka	Tidak Ada	Keluar	18:30:19	18:30:22	3
5	Terbuka	Tidak Ada	Masuk	18:30:25	18:30:27	2
6	Tertutup	Tidak Ada	Masuk	18:30:29	18:30:32	3

7	Tertutup	Ada	Masuk	18:35:06	18:35:08	2
8	Tertutup	Ada	Masuk	18:35:11	18:35:14	3
9	Terbuka	Ada	Keluar	18:35:19	18:35:21	3
10	Terbuka	Tidak Ada	Keluar	18:35:25	18:35:27	2
11	Terbuka	Tidak Ada	Masuk	18:35:32	18:35:35	3
12	Tertutup	Tidak Ada	Masuk	18:35:40	18:35:43	3
Rata-rata						2,5



Gambar 4. 13 Pengujian Sistem IoT

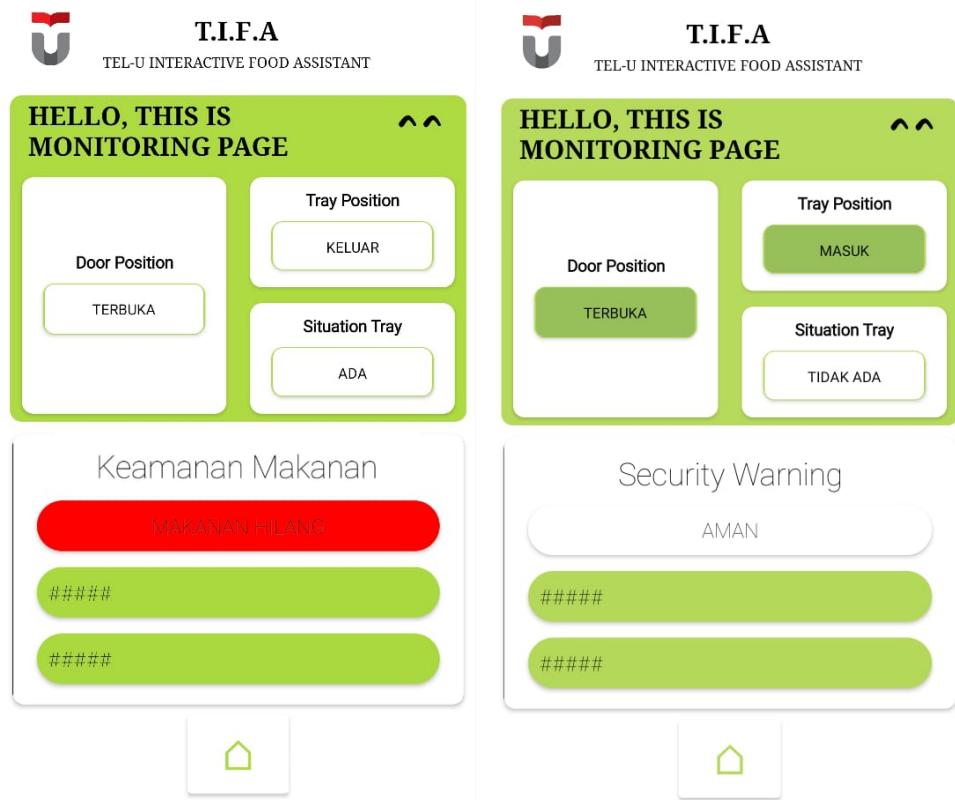
Dari hasil pengujian sebanyak 12 sampel, didapatkan bahwa rata-rata delay sistem pemantauan jarak jauh adalah 2,5 detik, Nilai ini menunjukkan bahwa sistem memiliki kemampuan untuk mengirim dan menyimpan data ke Firebase dalam waktu yang cukup responsif untuk kebutuhan monitoring berbasis IoT,

#### 4.1.6 Pengujian Keamanan Makanan

Pengujian ini bertujuan untuk mengevaluasi tingkat keamanan sistem pintu otomatis berbasis Internet of Things (IoT) yang telah dirancang. Fokus utama pengujian adalah memastikan bahwa sistem mampu merespons dengan tepat terhadap kondisi akses yang sah maupun tidak sah, dengan memantau

status pintu, sensor infra merah (IR). Data pengujian diperoleh dengan mencatat waktu kejadian di serial monitor sebagai catatan dari sisi perangkat (transmitter) dan waktu pencatatan di Firebase sebagai bukti penyimpanan di server (receiver).

Tabel 4. 6 Pengujian delay sistem keamanan pintu



Gambar 4. 14 Tampilan IoT Keamanan Makanan

Dari hasil pengujian sebanyak 12 sampel, didapatkan bahwa rata-rata delay sistem pemantauan jarak jauh adalah 2,4 detik, Nilai ini menunjukkan bahwa sistem memiliki kemampuan untuk mengirim dan menyimpan data ke Firebase dalam waktu yang cukup responsif untuk kebutuhan monitoring berbasis IoT,

## 4.2 Analisis

Pada bagian ini akan menjelaskan keterkaitan antara data dan hasil temuan yang diperoleh selama melakukan pengujian,

### 4.2.1 Analisis fungsi *Limit Switch KW10-ZIP* sebagai Pengaman dan Pengontrol Posisi

Pada bagian ini dua *limit switch* dipasang secara berdekatan di bagian sisi kanan robot (jika dilihat dari tampak depan) sebagai tanda gerak mekanis, Berdasarkan hasil pengujian, *limit switch* ini mampu memberikan respon yang cukup baik, Ketika pintu bergerak dan terkena menekan *limit switch*, motor akan berhenti dan saat dilepas, motor kembali bergerak, Pola

kerja ini memastikan bahwa motor tidak melampaui batas gerak yang telah dirancang, sehingga setiap pergerakan tetap terkendali,

Limit switch berfungsi sebagai pengaman untuk mencegah pergerakan yang berlebihan dan dapat merusak komponen mekanik sistem, Selain itu, perangkat ini juga berperan penting dalam sinkronisasi tahapan kerja otomatis, Pintu akan terbuka secara otomatis jika robot telah sampai pada meja tujuan, dan pintu akan menutup kembali jika *tray* makanan sudah benar-benar masuk,

Selain memiliki dua fungsi sebagai sebagai penentu batas gerak dan pengatur urutan operasi, *Limit switch* dapat berperan sebagai solusi tambahan ketika kontrol PID tidak cukup akurat, khususnya dalam menjamin posisi akhir atau batas gerak, Dalam hal ini, limit switch memberikan sinyal pasti saat titik batas tercapai, sehingga membantu menjaga keamanan dan ketepatan sistem, dan memastikan seluruh proses berjalan secara baik, presisi, dan sesuai urutan yang diharapkan,

#### 4.2.2 Analisis Integrasi Sistem Monitoring dan Kontrol IoT

Pada bagian ini sistem monitoring adalah hasil integrasi dari berbagai mikrokontroler dan sensor yang saling terhubung serta berkomunikasi satu sama lain, Selain berperan dalam menjalankan program pada pintu, ESP32 juga berperan dalam menerima dan mengirimkan status sistem ke firebase melalui raspberry pi, Dalam hal ini Raspberry Pi berfungsi sebagai komunikasi 2 arah untuk monitoring kondisi pintu, *tray* dan keadaan tray makanan dengan kondisi status yang akan diterima dari ESP32 dan Arduino Mega, Sementara itu Kodular digunakan sebagai platform monitoring pemantauan berbasis smartphone, Seluruh komponen ini beroperasi secara bersamaan dan *real-time*,

Dengan dukungan Firebase, pengguna dapat memantau kondisi pintu, *tray*, dan makanan secara langsung melalui perangkat seluler, Sistem ini juga mampu merespons perintah dari Raspberry Pi untuk menjalankan fungsi secara otomatis, tanpa memerlukan intervensi manual,

Keberhasilan dalam komunikasi ini menunjukkan bahwa sistem IoT yang dikembangkan tidak hanya mampu menjalankan kontrol otomatis, tetapi juga menyediakan fitur pemantauan yang responsif dan fleksibel, Hal ini secara signifikan meningkatkan efisiensi serta kenyamanan dalam pengoperasian robot pengantar makanan,

#### **4.2.3 Analisis Penggunaan Kontrol PID Pada Pergerakan Pintu**

Pengaplikasian kontrol PID dilakukan untuk mengetahui pengaruh variasi parameter PID terhadap performa pintu otomatis dengan menggunakan rotary encoder sebagai sensor feedback, Rotary encoder memiliki peran penting karena memberikan data posisi aktual pintu dalam bentuk pulsa digital yang dapat diolah secara real-time oleh mikrokontroler, Dengan feedback ini, kontrol PID dapat menghitung error secara presisi dan menyesuaikan pergerakan motor agar pintu bergerak sesuai setpoint,

Pada pengujian pertama dengan  $K_p=2.0$ ,  $K_i=0.05$ ,  $K_d=4.0$ , Gain proporsional yang lebih rendah membuat dorongan awal lebih lembut, sehingga waktu menuju setpoint sedikit lebih lama dibanding konfigurasi agresif.  $K_i$  kecil tetap menyapu error sisa secara bertahap, dan  $K_d$  tinggi memberi redaman kuat sehingga transisi halus, minim osilasi, serta overshoot kecil. Kombinasi ini cocok jika prioritasnya kenyamanan gerak (smooth) dan stabilitas, dengan trade-off respons yang agak lebih lambat.

Pada pengujian kedua dengan  $K_p=3.0$ ,  $K_i=0.2$ ,  $K_d=4.0$ ,  $K_i$  yang lebih besar mempercepat hilangnya steady-state error, tetapi meningkatkan risiko overshoot karena integrator “mendorong” sistem melewati setpoint sebelum diredam.  $K_d$  tinggi menahan osilasi berkepanjangan, sehingga sistem tetap cepat stabil. Hasilnya: tracking cepat dan akurat di akhir, namun ada puncak sementara (overshoot) saat mendekati  $178^\circ$ —cocok bila ketepatan akhir sangat diprioritaskan dan overshoot kecil masih dapat ditoleransi.

Pada pengujian ketiga dengan  $K_p=3.0$ ,  $K_i=0.05$ ,  $K_d=1.0$ , Dengan  $K_d$  yang diturunkan, efek redaman berkurang. Respons awal masih cepat

(karena K<sub>p</sub> moderat) dan error sisa tetap disapu Ki kecil, tetapi mendekati setpoint cenderung muncul riak (*underdamped*) kecil/ayunan singkat sebelum tenang. Ini memberi rasa respons yang lebih “gesit”, namun transisi kurang sehalus konfigurasi dengan K<sub>d</sub> besar.

Pada pengujian keempat dengan K<sub>p</sub>=1.5, K<sub>i</sub>=0.02, K<sub>d</sub>=1.0, Semua gain relatif kecil, sehingga keseluruhan respons paling “kalem”. Laju naik menuju 178° lambat, dan proses turun ke 1,5° juga lebih lama. Ki rendah membuat penghilangan error sisa berjalan pelan, sementara K<sub>d</sub> rendah membatasi redaman. Kelebihannya hampir tanpa overshoot, tetapi kekurangannya adalah waktu pencapaian target cukup panjang dan tracking perubahan cepat kurang presisi.

Pada pengujian ketiga dengan K<sub>p</sub>=3.0, K<sub>i</sub>=0.05, K<sub>d</sub>=4.0, K<sub>p</sub> memberi dorongan awal yang sigap, Ki kecil menghapus error sisa tanpa memicu overshoot berarti, dan K<sub>d</sub> besar meredam perubahan mendadak sehingga transisi mulus serta stabil. Umumnya menghasilkan waktu capai singkat, minim osilasi, dan posisi akhir presisi—tepat sebagai acuan performa.

Secara keseluruhan, Kesimpulannya, dari lima kombinasi parameter yang diuji, konfigurasi K<sub>p</sub>=3.0, K<sub>i</sub>=0.05, K<sub>d</sub>=4.0 paling seimbang untuk pintu otomatis dengan setpoint 178° (buka) dan 1,5° (tutup): responnya cepat, redaman baik, overshoot minim, dan error akhir kecil. Menurunkan K<sub>p</sub> (mis. 2.0) membuat gerak lebih lambat namun stabil; menaikkan K<sub>i</sub> (mis. 0.2) mempercepat hilangnya steady-state error tetapi cenderung memunculkan overshoot; menurunkan K<sub>d</sub> (mis. 1.0) mengurangi redaman sehingga muncul riak kecil menjelang setpoint; sedangkan semua gain kecil (K<sub>p</sub>=1.5, K<sub>i</sub>=0.02, K<sub>d</sub>=1.0) menghasilkan respon paling halus namun lambat dan kurang presisi pada perubahan cepat. Dengan demikian, kombinasi baseline (3.0, 0.05, 4.0) direkomendasikan untuk operasi, sementara penyesuaian kecil dapat dilakukan sesuai prioritas: naikkan K<sub>i</sub> jika ingin presisi akhir lebih ketat (terima sedikit overshoot), atau naikkan K<sub>d</sub> jika ingin transisi lebih halus.

#### 4.2.4 Analisis Pengujian Keamanan Makanan

Pengujian ini dilakukan untuk mengevaluasi kinerja sistem keamanan makanan yang terintegrasi dengan sistem pemantauan pintu berbasis IoT. Fokus utama pengujian adalah memastikan kemampuan sistem dalam mendeteksi kondisi keberadaan makanan pada wadah (tray) menggunakan sensor infra merah (IR) serta mengklasifikasikan status keamanan makanan menjadi "Aman" atau "Makanan Hilang".

Berdasarkan data hasil pengujian, status keamanan makanan ditentukan oleh kombinasi dari Status IR dan Status Tray. Jika sensor IR mendeteksi keberadaan makanan dan posisi tray berada pada kondisi masuk, maka sistem mencatat "Aman". Sebaliknya, jika sensor IR tidak mendeteksi makanan pada saat tray berada di dalam, maka sistem mencatat "Makanan Hilang".

Hasil pengujian menunjukkan bahwa sistem mampu memberikan status keamanan secara real-time dengan rata-rata delay sebesar 3 detik dari saat data terekam di serial monitor hingga tersimpan di Firebase. Nilai delay ini sudah cukup baik untuk aplikasi pemantauan jarak jauh, karena informasi keamanan makanan dapat diakses pengguna hampir tanpa jeda signifikan.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil pengujian dan analisis menunjukkan bahwa seluruh komponen sistem, termasuk bagian sensor, kontrol, dan komunikasi, bekerja dengan baik dan sesuai dengan tujuan perancangan, Sensor rotary encoder yang digunakan sebagai feedback posisi pintu terbukti mampu membaca posisi akhir pintu secara akurat, Limit switch juga berfungsi dengan baik sebagai pengaman tambahan, menghentikan motor ketika pintu mencapai batas akhir untuk mencegah kerusakan mekanik,

Dari sisi komunikasi, implementasi komunikasi dua arah antara ESP32 dan Arduino Mega telah berjalan lancar, Dengan kondisi sensor, status tray, dan data status perintah saling bertukar secara real-time, sistem dapat merespon setiap perintah secara cepat dan sinkron, Hal ini penting untuk memastikan bahwa sistem penggerak pintu, tray, serta pusat kontrol sistem bekerja sama dengan baik,

Selain itu, sistem berhasil mengimplementasikan Internet of Things (IoT) untuk monitoring pintu menggunakan ESP32 yang terintegrasi dengan Raspberry Pi dan platform Kodular, Melalui sistem IoT ini, pengguna dapat melihat status pintu dan tray secara real-time dengan sistem IoT ini dari jarak jauh,

Secara keseluruhan, selain keberhasilan dari aspek sensor, komunikasi, dan IoT, analisis menunjukkan bahwa sistem kendali yang digunakan termasuk dalam kategori kendali linier, Hal ini terlihat dari hubungan input-output yang proporsional, algoritma PID digital yang bekerja optimal pada sistem linier, parameter sistem yang konstan selama operasi. Dengan demikian, sistem pintu otomatis yang dikembangkan tidak hanya berfungsi sesuai perancangan, tetapi juga memenuhi karakteristik sebagai sistem kendali linier,

## 5.2 Saran

Meskipun sistem pintu otomatis yang telah dikembangkan untuk robot pengantar makanan telah menunjukkan kinerja yang baik, terdapat beberapa saran pengembangan yang dapat dipertimbangkan ke depannya untuk meningkatkan keandalan dan fleksibilitas sistem,

- Perbaikan metode kontrol pid untuk gerakan yang lebih halus

Meskipun teknik PID yang digunakan sudah mampu mengontrol pergerakan pintu dengan baik, diperlukan kalibrasi parameter PID yang lebih baik atau penggunaan variasi kontrol PID adaptif, Tujuannya adalah untuk memperhalus respons motor, mengurangi overshoot, dan mengurangi hentakan dan getaran saat mendekati posisi target, Penyempurnaan ini memungkinkan pintu bergerak dengan lebih mudah dan berhenti tepat di posisi akhir tanpa meningkatkan beban mekanis,

- Perbaruan prototipe

Prototipe yang masih terbatas dan kurang sesuai membuat pergerakan pintu menjadi kurang halus, serta memengaruhi kecepatan yang kurang stabil, maka dari itu diperlukan untuk membuat prototipe yang lebih baik, seperti mengganti pintu dengan bahan yang tidak mudah lentur, serta mengganti gear yang berbahan padat dan bentuk yang sesuai,

- Pengembangan terhadap aplikasi monitoring

Saat ini aplikasi hanya dapat menampilkan status pintu, Untuk pengembangan selanjutnya, disarankan agar sistem dilengkapi fitur kontrol jarak jauh melalui IoT (Internet of Things), sehingga pengguna dapat membuka atau menutup pintu secara manual dari jarak jauh jika diperlukan,

- Memperbarui desain sistem yang mendukung kemudahan perawatan

Untuk penggunaan jangka panjang, penting memastikan bahwa komponen dapat diakses dan diganti dengan mudah apabila terjadi kerusakan, Oleh karena itu, pengembangan selanjutnya disarankan merancang struktur dengan fleksibilitas yang baik agar proses perbaikan dan pemeliharaan tidak menyulitkan pengguna,

## DAFTAR PUSTAKA

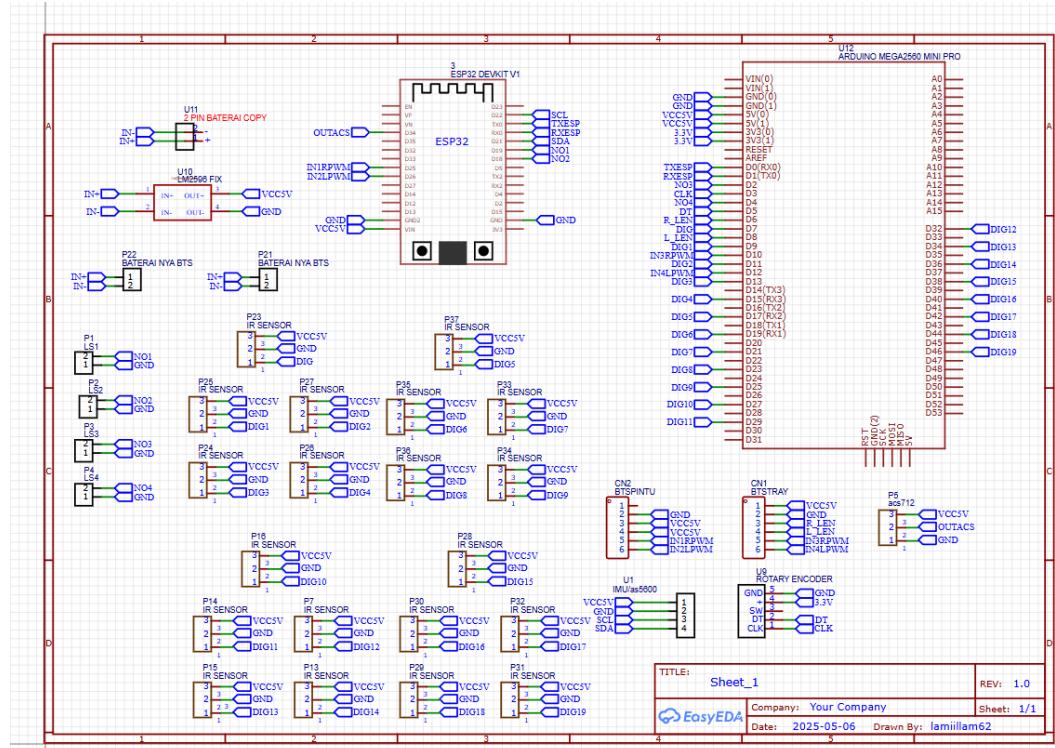
- [1] AVERINA FIDELIA AQILAH DIZARAMADHANI, "RANCANG BANGUN SISTEM MEKATRONIK KABINET PENYIMPANAN BATERAI PADA SISTEM PENUKARAN BATERAI OTOMATIS UNTUK UNMANNED AERIAL VEHICLE," Universitas Telkom, 2023.
- [2] F. T. Elektro and U. Telkom, *BUKU TUGAS AKHIR CAPSTONE DESIGN Mobilisasi Robot Pengantar Makanan Berbasis Odometry dan QR Detection*. 2023.
- [3] P. Raja, Dr. S. Kumar, D. S. Yadav, and Dr. T. Singh, "The Internet of Things (IOT): A Review of Concepts, Technologies, and Applications," *International Journal of Information technology and Computer Engineering*, no. 32, pp. 21–32, 2023, doi: 10.55529/ijitc.32.21.32.
- [4] E. Sopyan, D. Suarna, M. Harun Ashar, and Mustakim, "Rancang Bangun Robot Pengantar Obat dan Makanan Pasien Berbasis Internet of Things," *Bulletin of Information Technology (BIT)*, vol. 4, no. 3, pp. 344–352, 2023, doi: 10.47065/bit.v4i3.839.
- [5] A. Wardhana *et al.*, "Arsitektur dan standarisasi internet of things (iot)," *E-Commerce Dan Internet of Things (iot)*, no. May, pp. 197–210, 2023.
- [6] Y. A. Furima, J. P. P. Naibaho, and C. D. Suhendra, "Aplikasi Belajar Dan Bermain Untuk Anak Usia Dini Menggunakan Kodular," *JISTECH: Journal of Information Science and Technology*, vol. 11, no. 1, pp. 47–58, 2023, doi: 10.30862/jistech.v11i1.63.
- [7] M. I. Fahmi, J. Rohman, and E. Kuncoro, "Rancang Bangun Sistem Kendali Robot Tank Berbasis Internet of Things ( lot )," *Jurnal Telkommil*, pp. 1–7, 2020, [Online]. Available: <http://journal.poltekad.ac.id/index.php/kom/article/view/47>
- [8] A. Malewar, "PuduTech introduced BellaBot, a cute meowing food delivery robot." Accessed: Dec. 30, 2024. [Online]. Available: [https://www.inceptivemind.com/pudutech-bellabot-holabot-cute-meowing-food-delivery-robot/11193/#google\\_vignette](https://www.inceptivemind.com/pudutech-bellabot-holabot-cute-meowing-food-delivery-robot/11193/#google_vignette)
- [9] Charlotte Hu, "How robots are actually helping in restaurants | Popular Science." Accessed: Dec. 30, 2024. [Online]. Available: <https://www.popsci.com/technology/robots-in-restaurants/>
- [10] Y. Kristyawan and A. D. Rizhaldi, "An Automatic Sliding Doors Using RFID and Arduino," *International Journal of Artificial Intelligence & Robotics (IJAIR)*, vol. 2, no. 1, pp. 13–21, 2020, doi: 10.25139/ijair.v2i1.2706.

- [11] ShelfGenie, "What is a Lazy Susan? | ShelfGenie." Accessed: Dec. 24, 2024. [Online]. Available: [https://www-shelfgenie-com.translate.goog/blog/solutions-designs/lazysusan/?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=id&\\_x\\_tr\\_hl=id&\\_x\\_tr\\_pto=sge](https://www-shelfgenie-com.translate.goog/blog/solutions-designs/lazysusan/?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=sge)
- [12] Charles, "What is Lazy Susan Cabinet? - VENACE." Accessed: Dec. 24, 2024. [Online]. Available: <https://www.venace.com/id/lemari-susan-malas/>
- [13] L. Khakim, Sunarno, Sugiyanto, S. Siahaan, J. Simanjuntak, and D. Lumbantoruan, "Modern Control Systems" by Richard C. Dorf and Robert H. Bishop - Пощук Google," *Jurnal MIPA Unnes*, vol. 35, no. 2, pp. 38–49, 2022, [Online]. Available: <https://www.google.com/search?q=%22Modern+Control+Systems%22+by+Richard+C.+Dorf+and+Robert+H.+Bishop&oq=%22Modern+Control+Systems%22+by+Richard+C.+Dorf+and+Robert+H.+Bishop&aqs=edge..69i57.385j0j9&sourceid=chrome&ie=UTF-8>
- [14] Didi Widya Utama, "Sistem Kontrol Pada Modul Aliran Distribusi Bahan Baku Dengan Menggunakan Progammable Logic Controller," *Seminar Nasional Mesin dan Industri (SNMI3) 2007*, vol. Jurusan Te, pp. 1–12, 2007, [Online]. Available: <http://repository.untar.ac.id/1415/1/60 Didi Widya UNTAR.pdf>
- [15] R. F. D. Kuncahyo, Puput Wanarti Rusimamto, Endryansyah, and Nur Kholis, "Rancang Bangun Sistem Kendali Kecepatan Putaran Motor DC Menggunakan PID Controller," *Jurnal Teknik Elektro*, vol. 13, pp. 135–143, 2024.
- [16] V. N. Febrianto, "Aplikasi Kontrol PID untuk Pengaturan Putaran Motor DC pada Alat Pengepres Adonan Roti (Screw Conveyor)," *Jur. Tek. Elektro Fak. Tek. Univ. Brawijaya*, pp. 1–6, 2014.
- [17] R. Hudaya, "Kendali PID Digital: Implementasi Pada Sistem Tertanam menggunakan Metode Pengolahan Terdistribusi," *JTERA (Jurnal Teknologi Rekayasa)*, vol. 9, no. 2, p. 85, 2024, doi: 10.31544/jtera.v9.i2.2024.85-94.
- [18] Kifaya and C. Rijal, "Desain dan Implementasi Kontroler Proportional, Integral, Diferensial (PID) Digital dengan Human Machine Interface (HMI) pada Modul Trainer Analog Control ...," *Seminar Nasional Hasil Penelitian*, vol. 2017, pp. 49–54, 2018, [Online]. Available: <http://jurnal.poliupg.ac.id/index.php/snp2m/article/viewFile/213/151>
- [19] P. Saka Gilap Asa and S. Priyambodo, "Sistem Pembelajaran Kontrol Pid (Proporsional Integral Derivatif) Pada Pengatur Kecepatan Motor Dc Pid(Proportional Integral Derivative) Control Learning System on Dc Motor Speed Controller," *Jurnal Elektrikal*, vol. 3, no. 1, pp. 72–77, 2016.
- [20] S. Widodo and M. Baidlowi, "Rancang Bangun Sensor Rotary Encoder Sebagai Alat Pengendali Gerakan Azimuth Prototype Dudukan Meriam," *Jurnal Elektrosista*, vol. 11, no. 1, pp. 1–9, 2023, doi: 10.63824/jtep.v11i1.147.

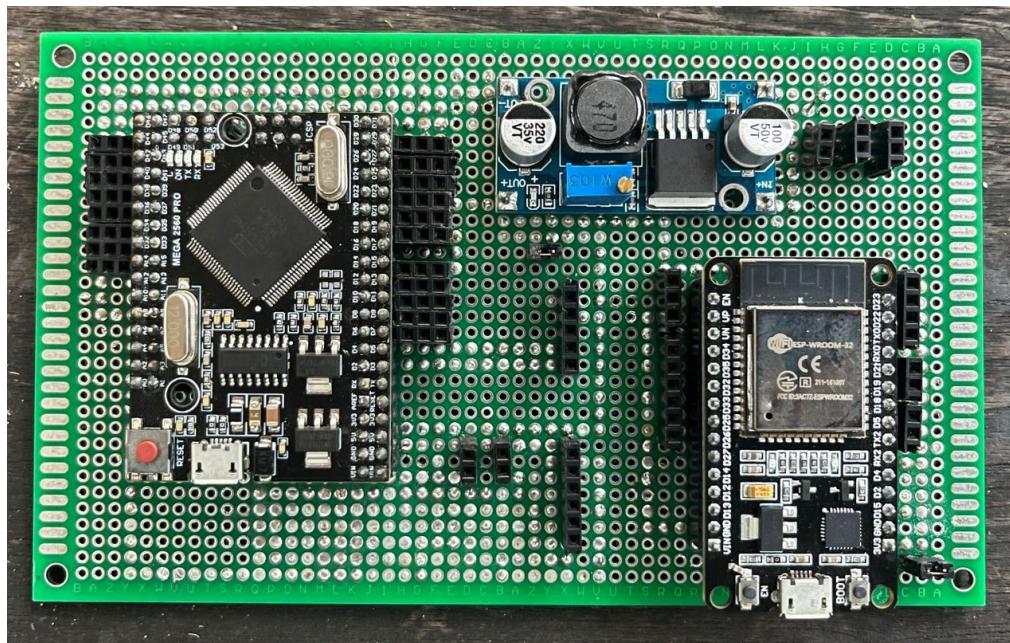
- [21] I. Akbar, N. Ismail, and T. D. Rachmilda, “Design of Angle Position and Temporary Velocity Detector Using Rotary Encoder KY-040,” *SENTER 2019: Seminar Nasional Teknik Elektro 2019*, no. November 2020, pp. 287–293, 2020.

## LAMPIRAN

### 1. Pinout PCB



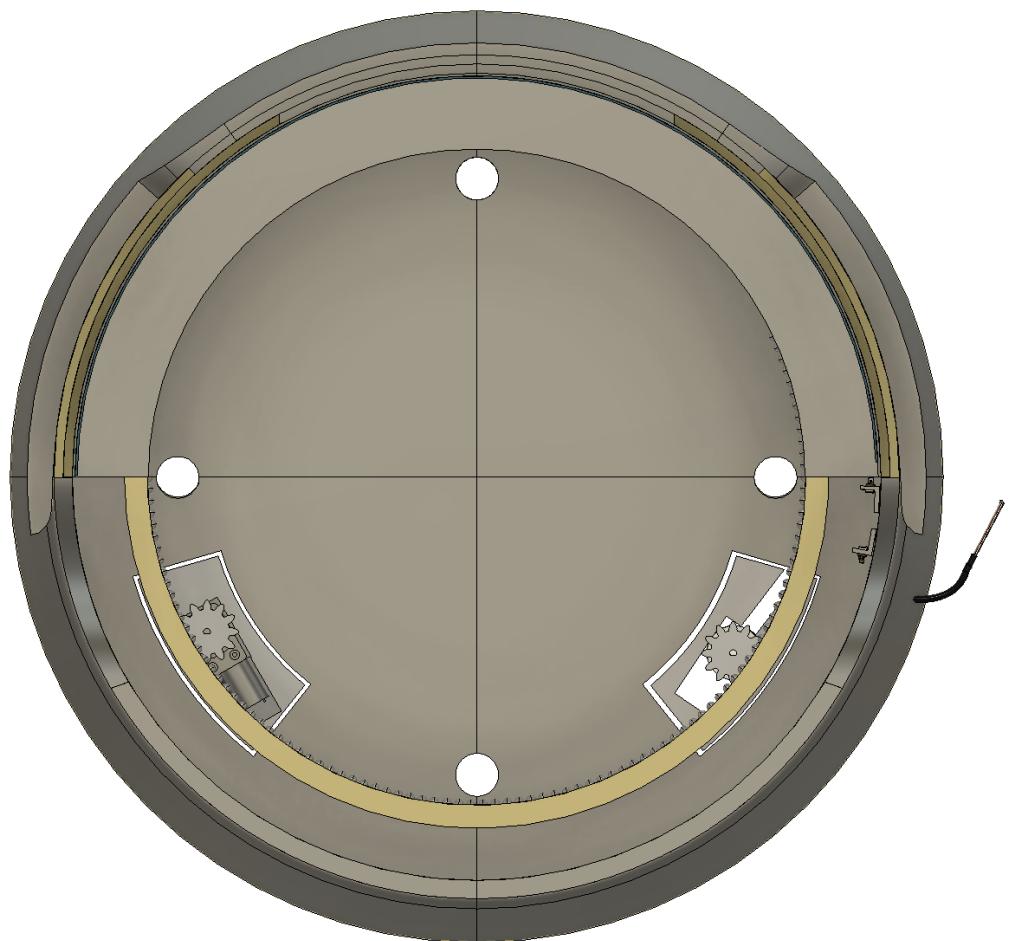
### 2. PCB



3. Robot tampak depan



4. Penempatan Sensor dan motor pada pintu



5. Codingan Pintu

```
#define LIMIT_BUKA 18
#define LIMIT_TUTUP 19
#define RPWM 25
#define LPWM 26
#define RXD2 16
#define TXD2 17
#define ENCODER_A 21
#define ENCODER_B 22

char data_mega;
bool state = false, command_buka = false, manual_tutup = false;
char trayState = '0', selesai = '0';
bool irDetected = false;

double Kp = 3.0, Ki = 0.05, Kd = 4.0;
double setpoint = 0.0, input, output, error, lastError = 0, integral = 0;
const double outMin = -60, outMax = 100, tolerance = 0.5;

volatile long encoderCount = 0;
volatile int lastEncoded = 0;
const long maxPulse = 13899;
bool pidComplete = false;

// const int pinADC = 34;
// const float VREF = 3.3;
// const float ADC_MAX = 4095.0;
// const float dividerRatio = 16.0;
// const float voltMin = 49.2;
// const float voltMax = 53.0;
// const float stepVolt = 0.1;
// const float stepPercent = 100.0
// int persenBulat;
// float batteryVoltage;

void stopMotor();
void runMotor();
void computePID();
void fullMonitor();
// void bacaTegangan();
void setPointBuka();
void setPointTutup();
void IRAM_ATTR updateEncoder();
```

```

void IRAM_ATTR updateEncoder() {
    int MSB = digitalRead(ENCODER_A);
    int LSB = digitalRead(ENCODER_B);
    int encoded = (MSB << 1) | LSB;
    int sum = (lastEncoded << 2) | encoded;

    if (sum == 0b1101 || sum == 0b0100 || sum == 0b0010 || sum == 0b1011) encoderCount++;
    if (sum == 0b1110 || sum == 0b0111 || sum == 0b0001 || sum == 0b1000) encoderCount--;

    lastEncoded = encoded;
}

void setup() {
    Serial.begin(115200);
    Serial2.begin(9600, SERIAL_8N1, RXD2, TXD2);

    pinMode(LIMIT_BUKA, INPUT_PULLUP);
    pinMode(LIMIT_TUTUP, INPUT_PULLUP);
    pinMode(RPWM, OUTPUT);
    pinMode(LPWM, OUTPUT);
    pinMode(ENCODER_A, INPUT_PULLUP);
    pinMode(ENCODER_B, INPUT_PULLUP);

    lastEncoded = (digitalRead(ENCODER_A) << 1) | digitalRead(ENCODER_B);
    attachInterrupt(digitalPinToInterrupt(ENCODER_A), updateEncoder, CHANGE);
    attachInterrupt(digitalPinToInterrupt(ENCODER_B), updateEncoder, CHANGE);

    stopMotor();
    Serial.println("Ketik '1' = BUKA pintu, '2' = TUTUP pintu manual, '0' = STOP motor");
}

void loop() {
    input = ((float)encoderCount / maxPulse) * 180.0;
    input = constrain(input, 0, 180);

    fullMonitor();
}

```

```

if (Serial.available()) {
    char cmd = Serial.read();
    if (cmd == '1') setPointBuka();
    else if (cmd == '2') setPointTutup();
    else if (cmd == '0') { stopMotor(); state = false; command_buka = false; manual_tutup = false; setpoint = input; }
}

if (Serial2.available()) {
    data_mega = Serial2.read();
    if (data_mega == '1') { setPointTutup(); traystate = '0'; }
}

if (state) {
    if (command_buka) {
        if (digitalRead(LIMIT_BUKA) == LOW || pidComplete) {
            stopMotor();
            traystate = '1';
            Serial2.write(trayState);
            state = false; command_buka = false; pidComplete = false;
        } else {
            computePID();
            runMotor();
            if (fabs(180.0 - input) <= tolerance) pidComplete = true;
        }
    }
    else if (manual_tutup || (!command_buka && !manual_tutup)) {
        if (digitalRead(LIMIT_TUTUP) == LOW || pidComplete) {
            stopMotor();
            state = false; selesai = '1'; pidComplete = false;
            if (fabs(input) <= tolerance) encoderCount = 0;
        } else {
            computePID();
            runMotor();
            if (fabs(input) <= tolerance) pidComplete = true;
        }
    }
}

void computePID() {
    error = setpoint - input;
    integral += error;
    integral = constrain(integral, -300, 300);
    double derivative = error - lastError;
    output = (Kp * error) + (Ki * integral) + (Kd * derivative);
}

```

```

}

void runMotor() {
    int pwm = abs(output);
    if (pwm < 35 && pwm > 0) pwm = 35;

    if (output > 0) { analogWrite(RPWM, pwm); analogWrite(LPWM, 0); }
    else if (output < 0) { analogWrite(RPWM, 0); analogWrite(LPWM, pwm); }
}

void stopMotor() {
    analogWrite(RPWM, 0);
    analogWrite(LPWM, 0);
}

void fullMonitor() {
    // bacaTegangan(); // dinonaktifkan
    Serial.print(command_buka ? '1' : '0'); Serial.print(";");
    Serial.print(irDetected ? "1" : "0"); Serial.print(";");
    Serial.print(trayState); Serial.print(";");
    Serial.println(selesai);
    // Serial.print(";");
    Serial.print(persenBulat);
}

void setPointBuka() {
    command_buka = true;
    manual_tutup = false;
    state = true;
    selesai = '0';
    setpoint = 178.0;
    integral = 0;
    pidComplete = false;
}

void setPointTutup() {
    manual_tutup = true;
    command_buka = false;
    state = true;
    selesai = '0';
    setpoint = 1.5;
    integral = 0;
    pidComplete = false;
}

//void bacaTegangan() {
//    int adcValue = analogRead(pinADC);
//    float voltageADC = adcValue * VREF / ADC_MAX;
//    batteryVoltage = voltageADC * dividerRatio;
//    float roundedVoltage = floor(batteryVoltage * 10) / 10.0;
//    float rawPercent = (roundedVoltage - voltMin) / stepVolt * stepPercent;
//    rawPercent = constrain(rawPercent, 0.0, 100.0);
//    persenBulat = round(rawPercent);
//}
}

```