IMPLEMENTATION AND ANALYSIS OF CONNECTOR WIRE TESTING MACHINE BASED ON PID AND VISUAL INTERFACE

Diono, S.Tr.T., M.Sc.[1], Indra Mora[2]

[1]Politeknik Negeri Batam
Department of Electrical Engineering
Jl. Ahmad Yani, Batam Centre, Batam 29461, Indonesia
E-mail: diono@polibatam.ac.id.ac

Abstract

In the era of industry 4.0, automation in quality testing process has become a crucial factor in ensuring production consistency, accuracy, and efficiency. This research develops a Connector Wire Testing Machine that integrates a load cell sensor, a servo motor control system based on a Proportional - Integral-Derivative (PID) controller, a real time visual interface developed in C#, and a SQL Server database for automated data logging. Load cell calibration was performed by comparing sensor readings with a digital reference scale over a load range of 1 kg to 3 kg. The results show that the measurement deviation remained below ±0,2%, indicating that the load cell provides accurate and consistent force measurement. Communication performance between the ESP32 and the C# graphical user interface was evaluated through bidirectional latency testing. The system achieved an average delay of 2,2 ms, which is well below the industrial response time threshold of 100 ms, demonstrating stable and responsive data transmission. Database performance testing showed a success rate of 100%, a duplicate rate of 0%, and a coefficient of variation (CV) of 11%, confirming reliable and stable real-time data processing. Furthermore, PID controller tuning resulted in optimal parameters of KP= 0.0025, Ki= 0.00015, and Kd= 0.003, which produced low overshoot (<1%), minimal steady-state error (<0.3%), and relatively short settling tomes across various setpoints. Overall, the developed system operates accurately, responsively, and reliably, supporting precise wire connector pull testing and robust data management for further analysis and continuous quality improvement.

Keywords: Connector Wire Testing Machine, Database, Load Cell, PID Control

1. Introduction

In the era of Industry 4.0, automation systems have become an essential part of production processes to improve efficiency, quality, and consistency of output. One of the key aspects in the electronics and manufacturing industries is testing the strength of cable connections at connectors, which is a crucial step in ensuring the reliability of connectivity systems in electronic products, automotive components, and industrial equipment [1].

Based on interviews conducted by the author with two employees of PT Simatelex Manufactory Batam, it was found that testing the connection strength between cables and connectors is still performed manually, using methods that lack standardization and carry a high risk of human error. The interviews were conducted with Mr. Iman and Mr. Fajar, who serve as IPQC (Inspector Quality Control) at PT Simatelex Manufactory Batam, on July 14, 2025, at 12:00 PM WIB.

Based on the interview results, it was found that the company needs to consistently perform tensile strength tests on cables connected to connectors. Currently, testing is still performed manually using a pull tester, where measurement results are read and recorded manually by the operator. This situation leads to inconsistent data, a slow testing process, and the risk of errors in documenting test results.

Both experts highlighted the need for an automatic tensile testing system capable of providing real-time test data, ensuring proper documentation, and featuring control mechanisms to maintain stability throughout the tensile testing process. In this context, the Proportional Integral Derivative (PID) algorithm is considered the appropriate method for maintaining the stability of the cable-pulling motor's motion, thereby allowing the tensile force to be applied in a controlled manner [2]. The implementation of PID control enables the speed and tensile force to remain stable throughout the test, resulting in more accurate, consistent, and reliable test data [3].

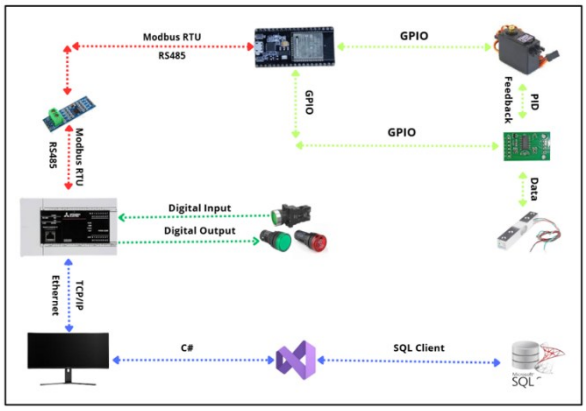
In addition to the control system, the need for a visual interface is also a critical aspect of system development.

By utilizing a C#-based visual interface, the system is capable of storing test results in a database. Given these conditions and practical needs, this research is aimed at designing and developing a machine for testing the strength of cable connections to connectors, based on PID control and a visual interface. The developed system is expected to accurately and stably control cable tensile force using a servo motor with PID control, display and store tensile strength test results digitally and in real-time, and improve efficiency, accuracy, and the quality of documentation in the cable connection testing process. With the introduction of this system, it is hoped that it will serve as an effective solution to improve the quality and productivity of the testing process in an industrial setting, as well as address the actual needs faced by PT Simatelex Manufactory Batam.

2. Research Methods

2.1. System Design

In this study, the author divided the system design into two aspects: the first is the system block diagram, and the second is the system design. The details are as follows:

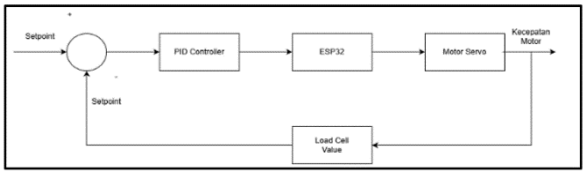


Picture 1:Diagram Block System

The PID-controlled Cable Connector Testing Machine with a visual interface is designed to automatically perform tensile strength testing on cable connectors with high precision and database-driven monitoring. The testing process begins with a load cell sensor that measures the tensile force as the cable is pulled by a servo motor. The measurement data is then sent as a feedback signal to the microcontroller, which acts as the main controller in the PID (Proportional-Integral-Derivative) system. This feedback value is compared to the predetermined tensile force setpoint; based on the difference between the two (error), the microcontroller calculates the PID correction signal to maintain a stable tensile force in line with the target value. The PID output signal is sent via the GPIO interface to the servo motor, which drives the cable-pulling mechanism. Changes in pulling force detected by the load cell are continuously sent back as feedback to the microcontroller, thereby forming a closed-loop control system that automatically maintains force stability throughout the testing process.

In addition, the microcontroller is connected to the Mitsubishi FX5U PLC via the Modbus RTU (RS-485) communication protocol. The PLC serves as the logic controller for the entire system and as an interface to the digital input and digital output devices. Digital inputs in the system include a START button to initiate the testing process and an Emergency Stop (EMG) button that stops the system immediately in the event of an emergency. Meanwhile, digital outputs are used to indicate test results, such as turning on a green light for a PASS status and a red light or buzzer for a FAIL status.

Once the testing process is complete, the PLC sends the test results to the GUI via a Modbus TCP/IP (Ethernet) network. The system runs a C#-based application that displays this data through an interactive visual interface, allowing operators to monitor tensile force values and test results in real time. All test results are then automatically saved to a SQL Server database, allowing the data to be utilized for documentation, statistical analysis, and quality tracing. With an integrated workflow between the PID Controller, PLC, and visual interface, this system is capable of providing efficient, stable, and accurate cable connection strength testing, as well as generating well-documented data.



Picture 2:PID System Diagram

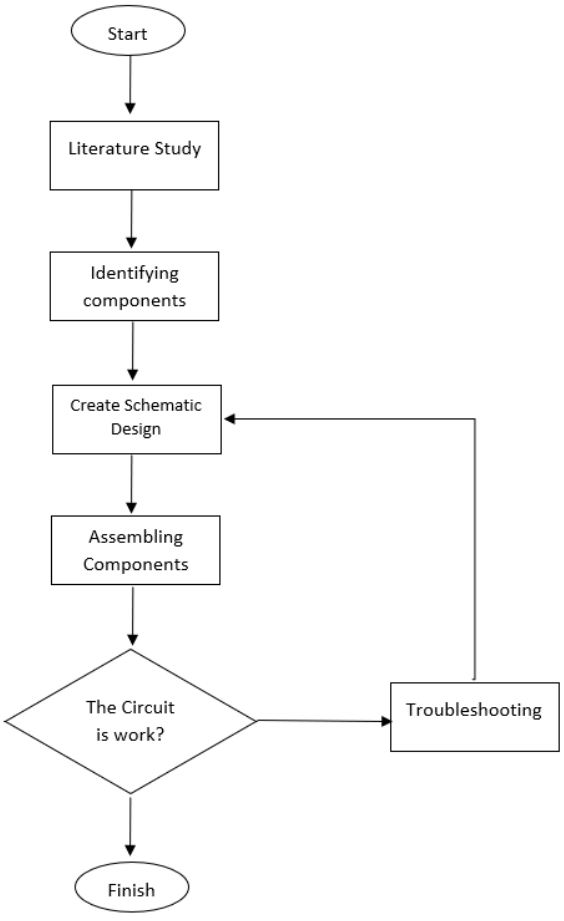
The system shown in the PID system diagram is a closed-loop control circuit based on the PID (Proportional-Integral-Derivative) method, implemented on an ESP32 microcontroller to regulate the movement of a servo motor during the cable connector tensile strength testing process. This system is designed to maintain a stable tensile force value in accordance with the specified setpoint. The system's operation begins with the input of a setpoint value—the target tensile force. This value is compared with the feedback signal obtained from the load cell sensor. The load cell measures the magnitude of the tensile force generated as the servo motor pulls the cable, and this value is then sent back to the ESP32 as input for the PID control system. The ESP32, which runs the PID algorithm, calculates the difference between the setpoint value and the actual value from the load cell (error). Based on this error value, the ESP32 generates a PID control signal containing proportional, integral, and derivative components. This signal is then used to regulate the servo motor, so that the pulling force can be dynamically and stably controlled according to the

target.

The servo motor mechanically pulls the cable, and the resulting continuous changes in force are measured by a load cell and sent back as feedback to the ESP32. This creates a closed-loop PID system that ensures the pulling force remains at the desired value despite changes in load or external disturbances. In this system, the PLC does not control the motor but serves for monitoring and communication, including reading test data to be forwarded to a visual interface or logging system. With the closed-loop PID system between the setpoint, ESP32, servo motor, and load cell, the system is able to maintain the stability of the pulling force with precision, ensuring that the testing process is accurate, responsive, and consistent according to the established parameters.

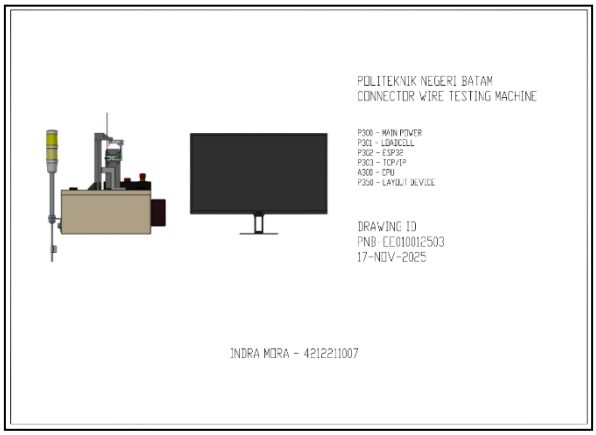
2.2. Electrical Design

The following is a flowchart of the electrical design process that the researcher will follow to ensure a more structured approach during the actual implementation. The steps and their explanations are as follows:

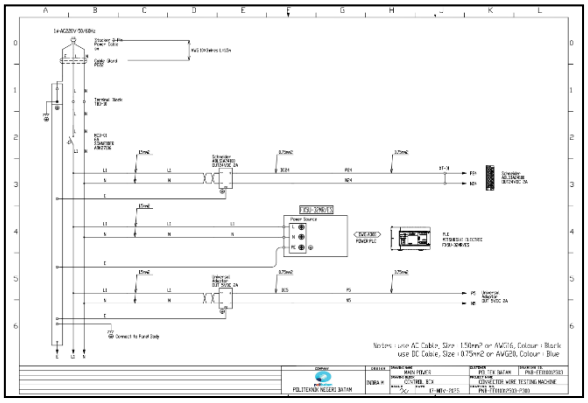


Picture 3:Electrical Design Flowchart

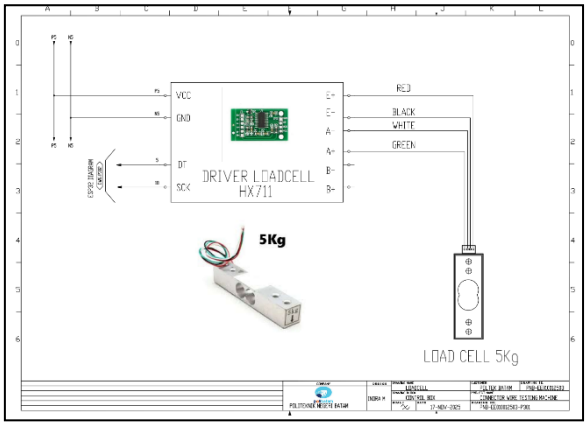
The electrical design phase begins with a literature review. The next step is to determine which components will be used. This is followed by creating a schematic design, which aims to facilitate the assembly process. The subsequent step involves assembling the components. Once all components have been assembled, a preliminary check is performed to verify whether the circuit is functioning properly. If not, troubleshooting is conducted, and the process is repeated starting from the schematic design step. However, if the circuit functions properly, the electrical design is considered complete. The following is the electrical design:



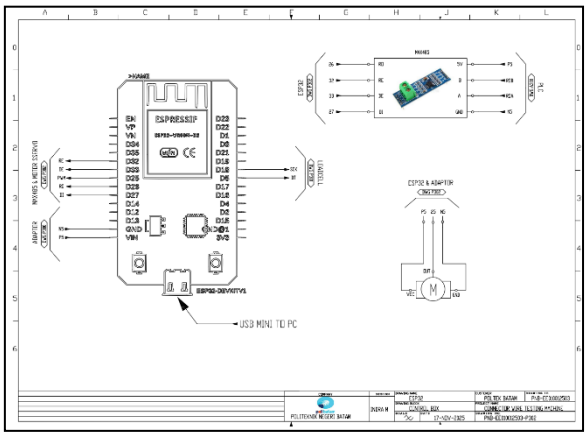
Picture 4:Electrical Design Cover



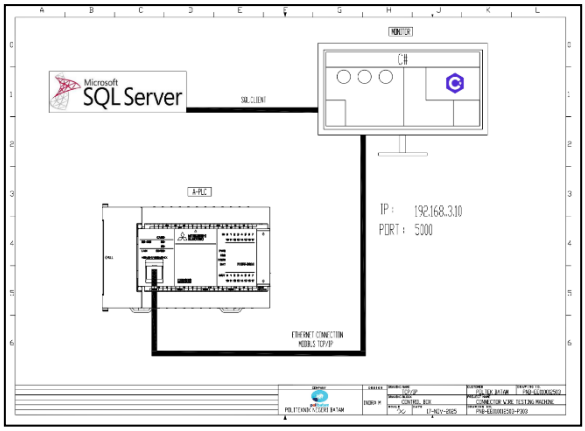
Picture 5:Electrical Power Design



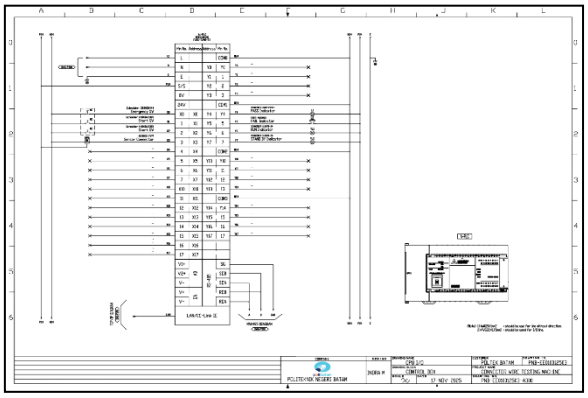
Picture 6:Electrical Loadcell Deisgn



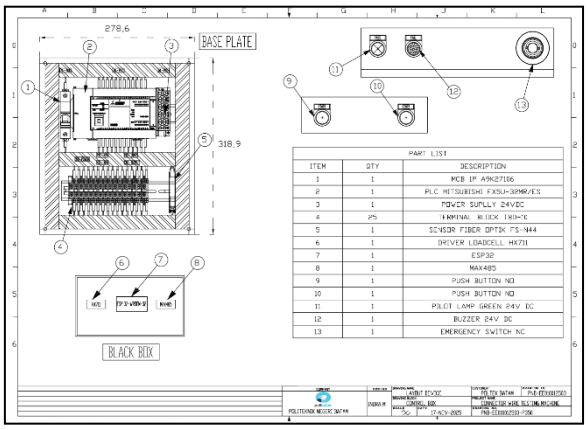
Picture 7:Eectrical ESP32 Design



Picture 8:Electrical TCP/IP Design



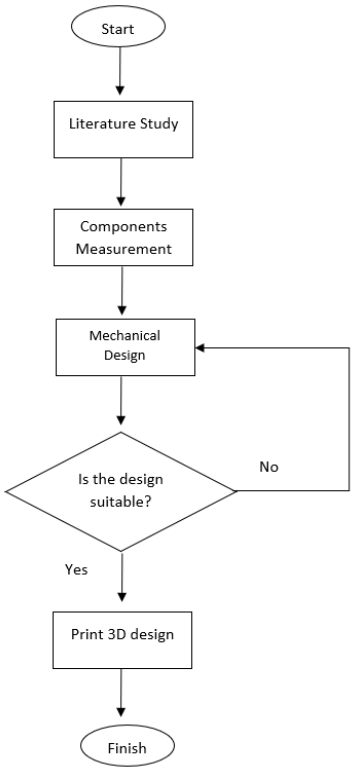
Picture 9:Electrical CPU I/O Design



Picture 10:Electrical Design Layout Device

2.3 Mechanical Design

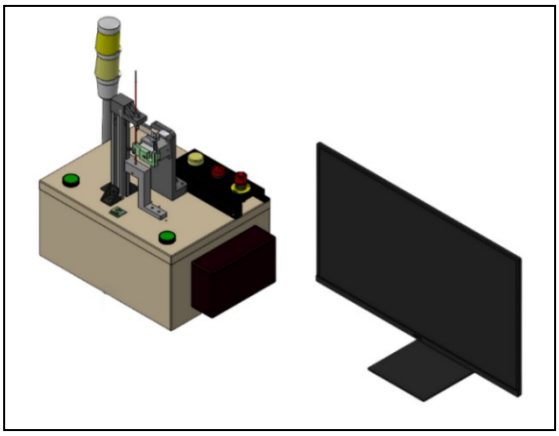
The following is a flowchart of the mechanical design process that the researcher will follow to ensure a more structured approach during the actual implementation. The steps and their explanations are as follows:



Picture 11:Mechanical Design Flowchart

In this design process, the researcher first conducts a literature review or searches for references, then measures and designs the necessary components. Next, these components are assembled and inspected to ensure they meet the desired specifications. Finally, 3D printing is performed, the components are integrated with other parts, and the project is completed.

The following is the mechanical design:



Picture 12:Mechanical Design

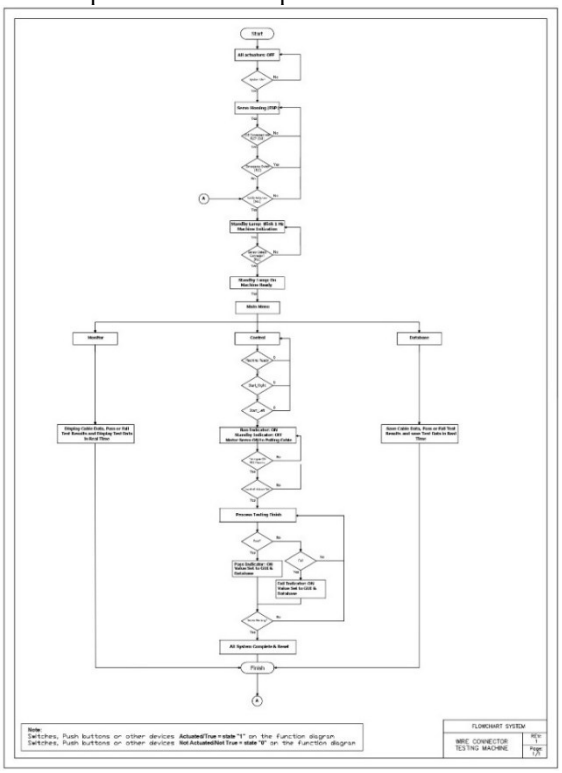
Description:

a) The controller box in this system is equipped with several key components that support the cable testing process, including:
   1) Motor Servo, berfungsi sebagai aktuator penarik kabel.
   2) Servo motor, which functions as a cable-pulling actuator.
   3) Load cell, used to measure the magnitude of the tensile force applied to the cable.
   4) ESP32, which acts as the control and data communication hub.
   5) The control buttons consist of a red button (emergency switch) and a green button (start button), mounted on the top of the control panel for ease of operation.
   6) There is also an indicator light to signal the system's status.
   7) The Mitsubishi PLC inside the control panel serves as the main controller, processing signals, executing control logic, and communicating with the ESP32 via the Modbus RTU (RS-485) protocol.

b) The System Layout shows the placement of components on the top surface of the box, including;
   1) The servo motor and load cell are centrally positioned to ensure balanced and measurable pulling force.
   2) The monitor is located on the side of the machine to facilitate monitoring and parameter settings

2.4 System Design

The following is a flowchart of the program design process that the researcher will follow to ensure a more structured approach during the actual implementation. The steps and their explanations are as follows:



Picture 13:Program Flowchart

The PID-Based Connector Wire Testing Machine with Visual Interface is an automated system designed to test the quality of cable connectors with high precision. The operating principle begins with system initialization, during which all actuators are ensured to be in the OFF state before the machine is turned on. The system displays a main menu with three primary options: the Settings Page for configuring test parameters such as PLC I/O and product datasets; the Home Page, which serves as the main dashboard displaying product calculations, servo motor position monitoring, and product datasets; and finally, the Database Page, which contains product data tables and a table monitoring product test results that updates in real-time once testing is complete.

The testing process begins by activating the motor, which is controlled by a PID controller, to reach the test position specified in the stored product dataset. A timer is activated to measure the duration of the test, and the system verifies that the connector is in the correct position before the test begins. If the test results meet the criteria (Pass), the Pass indicator lights up and the data is saved to the database. Conversely, if the test fails (Fail), the Fail indicator lights up and the data is still recorded for further analysis. Once the test is complete, the motor returns to its initial position and the system is reset to be ready for reuse.

The visual interface plays a crucial role in the system, particularly in helping operators control and monitor the testing process. This interface displays real-time information such as test status and test duration, and

stores all test result data in a database for quality control and reporting purposes. With PID control integration, this system is capable of performing tests with high accuracy, while the visual interface provides operational ease and monitoring. As a result, this Connector Wire Testing Machine not only improves testing reliability but also saves time compared to manual methods.

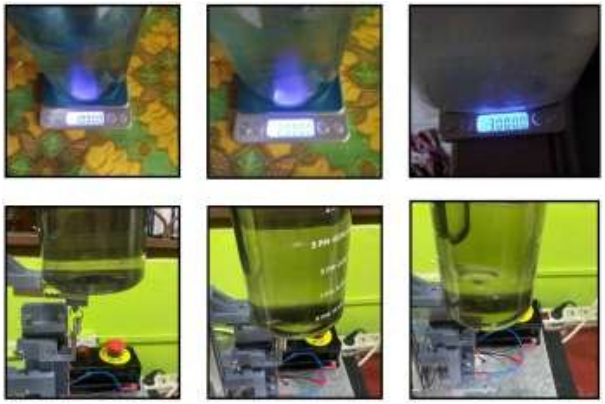
3. Result and Analysis

In this study, the author conducted several tests, which are described below.

No	Comparison Results				
	Timbangan (g)	Load Cell (g)	Error (%)	Standar Error (%)	Status
1	1.000	999	+0.1	±0.2	OK
2	1.000	1001	-0.1	±0.2	OK
3	1.000	999	+0.1	±0.2	OK
4	1.000	1000	0	±0.2	OK
5	1.000	1000	0	±0.2	OK
6	2.000	2000	0	±0.2	OK
7	2.000	2001	-0.05	±0.2	OK
8	2.000	2001	-0.05	±0.2	OK
9	2.000	1999	+0.05	±0.2	OK
10	2.000	2000	0	±0.2	OK
11	3.000	3002	-0.066	±0.2	OK
12	3.000	3001	-0.033	±0.2	OK
13	3.000	2999	+0.033	±0.2	OK
14	3.000	2999	+0.033	±0.2	OK
15	3.000	3000	0	±0.2	OK

3.1 Load cell calibration and accuracy testing

In this test, the accuracy and consistency of the load cell sensor readings used in the connector wire testing machine were evaluated. The test was conducted by comparing the load cell measurements with those of a standard digital scale. The test was performed using reference loads ranging from 1 kg to 3 kg.



Picture 14:Load Cell Calibration Testing

Picture 15: Load Cell 1 Kg Calibration Testing

Picture 16:Load Cell 2 Kg Calibration Testing

Picture 17:Load Cell 3 Kg Calibration Testing

The purpose of this test is to ensure that the load cell can accurately measure tensile force according to the actual value and can be used as a reference in the cable joint strength testing process. Each load point is tested to obtain readings from the digital scale (as a reference) and readings from the load cell sensor. Next, the percentage error is calculated to determine the extent to which the load cell sensor deviates from the actual value. The error value will be compared with the standard measurement error limit for load cells at PT Simatelex Manufactory Batam. The error calculation is performed using the following equation:

$$\text{Error (\%)} = \frac{\text{Meter Readings} - \text{Sensor Readings}}{\text{Meter Readings}} \times 100\% \quad (1)$$

Table 1:Loadcell Testing

From the load cell test table, it can be seen that the test results indicate the load cell readings have an average deviation of less than ±0.2% compared to the digital scale, and this value remains within the accuracy tolerance limits of the load cell as specified by PT Simatelex Manufactory Batam's industry standards. Therefore, it can be concluded that the sensor provides sufficiently accurate and consistent readings. Thus, the load cell can be relied upon as the primary instrument for measuring tensile force on the Connector Wire Testing Machine. Additionally, the application of calibration equations and tare procedures prior to testing can further enhance the accuracy of the readings, making the tensile strength test results more representative and reliable for further analysis.

3.2 Testing the latency and responsiveness of ESP32-GUI communication

This test was conducted to ensure that the Visual C# interface connected to the ESP32 operates accurately, responsively, and reliably as part of the cable tension

testing system. The first test was designed to measure real-time responsiveness, specifically the delay between the pull force signal read by the load cell and the position of the servo motor on the ESP32, and the time it takes for that data to be successfully displayed on the GUI. This delay value was compared to the internal standards of PT Simatelex Manufactory Batam to ensure that the system meets the minimum response time requirements.

The purpose of this testing is to assess how the system handles real-world conditions, maintains data integrity, and transmits data quickly. The results of all these tests will be summarized in an evaluation table to determine the reliability of the system's response in the cable-pulling test system.

```
==================================
LATENCY TEST #51
==================================
📤 Sending to C# (D15): 51
⏳ Waiting for C# response...
----------------------------------
📊 RESULTS:
----------------------------------
Sent (D15): 51
Received (D16): 151
C# processing (D18): 21 ms
Round-trip time (D17): 175 ms
Estimated one-way: 77 ms
----------------------------------
```

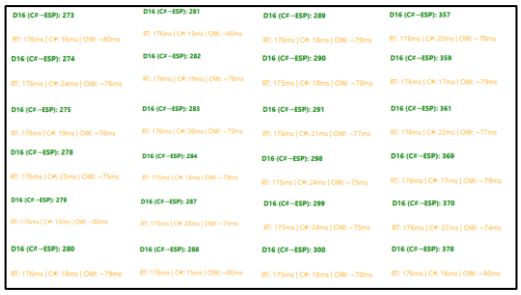
Picture 18:Latency Test Results

```
==================================
LATENCY TEST #52
==================================
📤 Sending to C# (D15): 52
⏳ Waiting for C# response...
----------------------------------
📊 RESULTS:
----------------------------------
Sent (D15): 52
Received (D16): 152
C# processing (D18): 28 ms
Round-trip time (D17): 176 ms
Estimated one-way: 74 ms
----------------------------------
```

Picture 19:Latency Test Results

```
==================================
LATENCY TEST #56
==================================
📤 Sending to C# (D15): 56
⏳ Waiting for C# response...
----------------------------------
📊 RESULTS:
----------------------------------
Sent (D15): 56
Received (D16): 156
C# processing (D18): 26 ms
Round-trip time (D17): 175 ms
Estimated one-way: 74 ms
----------------------------------
```

Picture 20:Latency Test Results



Picture 21:Latency Test Results

Table 2: Latency and responsiveness testing

No	Comparison Results				
	Responsivitas ESP32 to GUI (ms)	Load Cell (g)	Error (%)	Standar Error (%)	Status
1	77	78	1	<100 ms	OK
2	74	77	3	<100 ms	OK
3	74	78	4	<100 ms	OK
4	78	79	1	<100 ms	OK
5	79	77	2	<100 ms	OK
6	79	79	0	<100 ms	OK
7	76	80	4	<100 ms	OK
8	80	76	4	<100 ms	OK
9	77	78	1	<100 ms	OK
10	76	75	1	<100 ms	OK
11	78	80	2	<100 ms	OK
12	78	79	1	<100 ms	OK
13	77	80	3	<100 ms	OK
14	79	78	1	<100 ms	OK
15	78	75	3	<100 ms	OK
16	80	79	1	<100 ms	OK
17	77	73	4	<100 ms	OK
18	78	80	2	<100 ms	OK
19	78	79	1	<100 ms	OK
20	79	78	1	<100 ms	OK
21	73	77	4	<100 ms	OK
22	79	75	4	<100 ms	OK
23	73	75	2	<100ms	OK
24	80	78	2	<100 ms	OK
25	79	78	1	<100 ms	OK
26	77	79	2	<100 ms	OK
27	74	77	3	<100 ms	OK
28	79	79	0	<100 ms	OK
29	79	74	5	<100 ms	OK
30	77	80	3	<100 ms	OK
Total Delay			66 ms		
Average Delay			2,2 ms		

Based on the results of observations using software, the latency (delay) values were calculated using the following formula:

$$Latency = \frac{\sum_{i=1}^{N}\left(t_{receive,i} - t_{send,i}\right)}{N} \qquad (2)$$

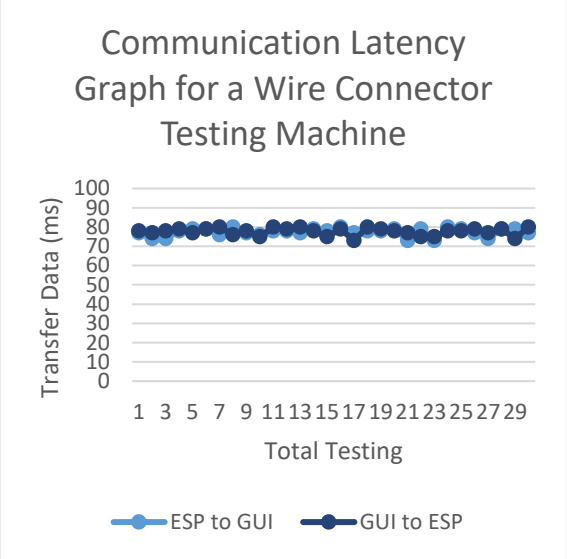
$$\text{Latency} = \frac{66}{30}$$

$$\text{Latency} = 2{,}2 \text{ ms}$$

Where $t_{receive,i}$ is the reception time, $t_{send,i}$ is the transmission time, and N is the number of data packets.

Based on 30 tests conducted to evaluate the responsiveness of two-way communication between the ESP32 and the C# graphical user interface (GUI). The primary parameters observed were the ESP32's response time to the GUI, the GUI's response time to the ESP32, and the total delay value calculated from the difference between the send time and receive time for each data transaction.

The test results show that all delay values fall within the range of 0–5 ms, with a total accumulated delay of 66 ms and an average delay of 2.2 ms. These values are well below the industry's maximum response limit (<100 ms), so the system can be classified as responsive and stable. To assess the stability of the response in the developed system, the following graph presents the delay patterns from the test results for 30 data transmission samples.



Picture 22:Latency Testing Chart (ms)

The graph above shows the stability of the two-way communication response time between the ESP32 and the GUI over 30 test runs. Both curves—ESP to GUI and GUI to ESP—exhibit a consistent pattern within a range of approximately 75–82 ms, without extreme fluctuations. This tight and relatively flat pattern indicates that the system is capable of maintaining stable and predictable transmission performance, in accordance with the established response standard limit (<100 ms). This consistency indicates that the communication mechanisms, data transmission processes, and message handling on both sides are functioning optimally and can be relied upon for continuous operation in industrial environments that require high.

3.3 Data processing and database performance testing

Data processing performance testing on SQL Server was conducted to evaluate the system's ability to process, store, and deliver test results quickly, consistently, and reliably while the Connector Wire Testing Machine is in operation. All pull test results are sent in real-time from the C#-based GUI interface to SQL Server, making the quality of data processing performance in the database a crucial factor in ensuring the overall reliability of the system. Testing was conducted by performing 50 data transmissions for each type of cable tested, and the response time for each transaction was recorded to obtain the Round Trip Time (RTT) value as an indicator of data processing speed.

A comprehensive analysis of the data transmission success rate and potential duplication was conducted by combining the results of all tests across the three types of cables. Additionally, a statistical analysis using the Coefficient of Variation (CV) for RTT values was performed on the entire dataset, ensuring that the resulting CV values comprehensively reflect the stability of SQL Server's response throughout the data transmission process.

Based on all the tests conducted on each subsystem and the integrated tests, a summary of the overall test results was compiled to assess the system's overall success rate. The summary of the overall system test results is shown in the following tablet:

Table 3: Overall system test results

TestID	CableCode	CableType	ConnectorType	Material	Diameter [mm]	TimingSetting [s]	PoundForce [lbf]	RoundTripTime [ms]	Status	Result	TestDateTime
1	A	Single	Crimping	Copper	0.1	60	3	0.175	OK	Good Product	12/12/2025 9:23
2	A	Single	Crimping	Copper	0.1	60	3	0.152	OK	Good Product	12/12/2025 9:25
3	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 9:27
4	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 9:29
5	A	Single	Crimping	Copper	0.1	60	3	0.156	OK	Good Product	12/12/2025 9:31
6	A	Single	Crimping	Copper	0.1	60	3	0.149	OK	Good Product	12/12/2025 9:33
7	A	Single	Crimping	Copper	0.1	60	3	0.149	OK	Good Product	12/12/2025 9:35

8	A	Single	Crimping	Copper	0.1	60	3	0.153	OK	Good Product	12/12/2025 9:37
9	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 9:39
10	A	Single	Crimping	Copper	0.1	60	3	0.195	OK	Good Product	12/12/2025 9:41
11	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 9:43
12	A	Single	Crimping	Copper	0.1	60	3	0.155	OK	Good Product	12/12/2025 9:45
13	A	Single	Crimping	Copper	0.1	60	3	0.169	OK	Good Product	12/12/2025 9:47
14	A	Single	Crimping	Copper	0.1	60	3	0.159	OK	Good Product	12/12/2025 9:49
15	A	Single	Crimping	Copper	0.1	60	3	0.164	OK	Good Product	12/12/2025 9:51
16	A	Single	Crimping	Copper	0.1	60	3	0.152	OK	Good Product	12/12/2025 9:53
17	A	Single	Crimping	Copper	0.1	60	3	0.150	OK	Good Product	12/12/2025 9:55
18	A	Single	Crimping	Copper	0.1	60	3	0.147	OK	Good Product	12/12/2025 9:57
19	A	Single	Crimping	Copper	0.1	60	3	0.118	OK	Good Product	12/12/2025 9:59
20	A	Single	Crimping	Copper	0.1	60	3	0.172	OK	Good Product	12/12/2025 10:01
21	A	Single	Crimping	Copper	0.1	60	3	0.196	OK	Good Product	12/12/2025 10:03
22	A	Single	Crimping	Copper	0.1	60	3	0.186	OK	Good Product	12/12/2025 10:05
23	A	Single	Crimping	Copper	0.1	60	3	0.149	OK	Good Product	12/12/2025 10:07
24	A	Single	Crimping	Copper	0.1	60	3	0.167	OK	Good Product	12/12/2025 10:09
25	A	Single	Crimping	Copper	0.1	60	3	0.155	OK	Good Product	12/12/2025 10:11
26	A	Single	Crimping	Copper	0.1	60	3	0.178	OK	Good Product	12/12/2025 10:13
27	A	Single	Crimping	Copper	0.1	60	3	0.113	OK	Good Product	12/12/2025 10:15
28	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 10:17
29	A	Single	Crimping	Copper	0.1	60	3	0.147	OK	Good Product	12/12/2025 10:19
30	A	Single	Crimping	Copper	0.1	60	3	0.147	OK	Good Product	12/12/2025 10:21
31	A	Single	Crimping	Copper	0.1	60	3	0.154	OK	Good Product	12/12/2025 10:23
32	A	Single	Crimping	Copper	0.1	60	3	0.147	OK	Good Product	12/12/2025 10:25
33	A	Single	Crimping	Copper	0.1	60	3	0.152	OK	Good Product	12/12/2025 10:27
34	A	Single	Crimping	Copper	0.1	60	3	0.153	OK	Good Product	12/12/2025 10:29
35	A	Single	Crimping	Copper	0.1	60	3	0.149	OK	Good Product	12/12/2025 10:31
36	A	Single	Crimping	Copper	0.1	60	3	0.158	OK	Good Product	12/12/2025 10:33
37	A	Single	Crimping	Copper	0.1	60	3	0.152	OK	Good Product	12/12/2025 10:35
38	A	Single	Crimping	Copper	0.1	60	3	0.120	OK	Good Product	12/12/2025 10:37
39	A	Single	Crimping	Copper	0.1	60	3	0.170	OK	Good Product	12/12/2025 10:39
40	A	Single	Crimping	Copper	0.1	60	3	0.175	OK	Good Product	12/12/2025 10:41
41	A	Single	Crimping	Copper	0.1	60	3	0.178	OK	Good Product	12/12/2025 10:43
42	A	Single	Crimping	Copper	0.1	60	3	0.152	OK	Good Product	12/12/2025 10:45
43	A	Single	Crimping	Copper	0.1	60	3	0.146	OK	Good Product	12/12/2025 10:47
44	A	Single	Crimping	Copper	0.1	60	3	0.154	OK	Good Product	12/12/2025 10:49
45	A	Single	Crimping	Copper	0.1	60	3	0.160	OK	Good Product	12/12/2025 10:51
46	A	Single	Crimping	Copper	0.1	60	3	0.178	OK	Good Product	12/12/2025 10:53
47	A	Single	Crimping	Copper	0.1	60	3	0.156	OK	Good Product	12/12/2025 10:55
48	A	Single	Crimping	Copper	0.1	60	3	0.176	OK	Good Product	12/12/2025 10:57
49	A	Single	Crimping	Copper	0.1	60	3	0.161	OK	Good Product	12/12/2025 10:59
50	A	Single	Crimping	Copper	0.1	60	3	0.157	OK	Good Product	12/12/2025 11:01

The test data obtained serves as evidence of the implementation of the Connector Wire Testing Machine designed in this final project. Testing was conducted 150 times using various sample types and cable diameters, demonstrating that the machine is capable of performing cable tensile tests on connectors automatically, stably, and consistently. The entire testing process resulted in an "OK" status, indicating that the mechanical system, control system, sensors, and data acquisition and recording system have been integrated and function properly as a single machine unit.

Table 4:Database Testing

NO	Cable Type	Cross-sectional Area (mm²)	Test Qty	Numbers of Entries	Number of Duplication	RTT Average (ms)	Result
1	A	0.10	50	50	0	0.158	OK
2	B	0.20	50	50	0	0.151	OK
3	C	0.30	50	50	0	0.154	OK
Success Rate			100%				OK
Duplicate Rate			0%				OK
Coefisient of Variation			11%				OK

Based on the results of the tests in the database test table, the values for incoming data and duplicate data were calculated using the following formula:

Meanwhile, to calculate the coefficient of variation from Table 9, the following formula was used:

$$\text{Success Rate} = \frac{\text{Input Data}}{\text{Total Test}} \text{ x } 100\% \quad (3)$$

$$\text{Success Rate} = \frac{150}{150} \text{ x } 100\%$$

$$\text{Success Rate} = \ 100\%$$

$$\text{Duplicate Rate} = \frac{\text{Duplicate Data}}{\text{Total Test}} \text{ x } 100\% \quad (4)$$

$$\text{Duplicate Rate} = \frac{0}{150} \text{ x } 100\%$$

$$\text{Duplicate Rate} = \ 0\%$$

To calculate the coefficient of variation from the

database test table, the following formula is used:

$$\mu = \mathbf{x_{i\,n}} \quad (5)$$

$$\mu = 0.158 + 0.151 + 0.1543$$

$$\mu = 0.154$$

Where μ is the average RTT, X_i is the sum of all RTT values, and n is the total number of products

$$\boldsymbol{\sigma} = \sqrt{\frac{\sum(\mathbf{x_{i-\mu}})^2}{\mathbf{n-1}}} \quad (6)$$

$$\sigma = \sqrt{\frac{0{,}048}{149}}$$

$$\sigma = 0.018$$

Where σ is deviation standard RTT, X_i is total RTT values, n is total samples and μ is RTT Average

$$\mathbf{CV} = \frac{\boldsymbol{\sigma}}{\boldsymbol{\mu}} \mathbf{x\ 100\%} \quad (7)$$

$$\mathbf{CV} = \frac{\mathbf{0.018}}{\mathbf{0.154}} \mathbf{x\ 100\%}$$

$$\mathrm{CV} = 11\%$$

Where CV is Coefficient of Variation, σ is the RTT standard deviation, dan μ is RTT Average

The following are the standard classifications for success rate, duplicate rate, and coefficient of variation at PT Simatelex Manufactory Batam:

Table 5:Parameters Standard

No	Parameter	Criteria	Criteria
1	*Success rate*	≥ 99%	Very Good
		95% – 98.99%	Good
		< 95%	Bad
2	*Duplicate Rate*	0%	Very Good
		0.01% – 1%	Good
		> 1%	Bad
3	*Coefficient of Variation* (CV)	< 10%	Very Good
		10% ≤ CV < 20%	Good
		≥ 20%	Bad

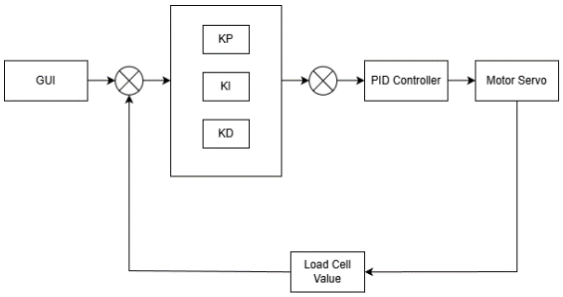
Based on the test results shown in the standard parameter table, all database system performance parameters including the success rate, duplicate rate, and coefficient of variation (CV) meet the "Good" quality category according to PT Simatelex Manufactory Batam standards. The success rate reached 100%, calculated from a total of 150 data entries compared to 150 total tests. This result falls into the "Very Good" category, which requires a minimum value of ≥ 99%. At the same time, the duplicate rate was recorded at 0%, meaning no duplicate data was found during the transmission and recording process. This condition also meets the "Very Good" category according to the standard, which sets the ideal duplicate rate at 0%.

Furthermore, the coefficient of variation (CV) analysis yielded a result of 11%, calculated from a standard deviation of RTT of 0.018 ms based on an average RTT of 0.154 ms. According to the CV classification, this value falls within the range of 10% ≤ CV < 20% and is therefore categorized as "Stable." This indicates that the variation in response time (RTT) among cable types is relatively low and remains within acceptable limits for system reliability.

Overall, the combination of a maximum success rate, a zero duplicates rate, and a stable RTT variation indicates that the machine's data transmission and logging system operates consistently and accurately. These results suggest that data integrity, transmission quality, and system performance stability are at a satisfactory level and suitable for use in industrial environments that require high accuracy.

3.4 Results of PID performance testing for cable tension stabilization

In this section, the researcher conducted tests on the PID system to determine the optimal PID controller settings for the servo motor system. The parameters tested were the control constants kp (proportional gain), ki (integral gain), and kd (derivative gain), which were selected to produce the best system response, such as minimal delay time, overshoot, controlled rise time, and low steady-state error. The method used is the trial-and-error method. This method is used because it provides flexibility in adjusting parameters directly based on observations of the system response. In this approach, the author gradually changes the values of Kp, Ki, and Kd, then analyzes changes in system characteristics, such as rise time, settling time, and system stability.



Picture 23:Diagram block system

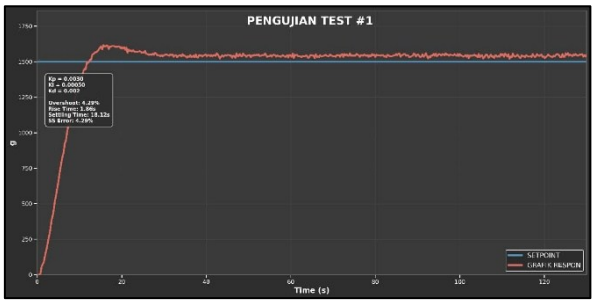
The system block diagram illustrates the control system used to regulate the speed of a servo motor. This system is designed to control the motor's speed based on a predetermined input value (setpoint). This input represents the desired load value, which is processed by the PID controller implemented on the ESP32 module. The PID controller generates an output in the form of a load value, which is then used to adjust the servo's rotation angle. The load value generated by the load cell directly affects the motor speed, in accordance with the calculations and adjustments performed by the

PID controller. The PID controller compares this actual speed data with the setpoint value to calculate the error difference.

3.4.1 Test results for the setpoint using a PID controller

At this stage, the author conducted tests aimed at obtaining baseline performance data for the servo motor system using a PID controller. The PID values were calculated through a tuning process using the Ziegel-Nichols method to determine accurate values for kp, ki, and kd. These three values were used to determine the optimal PID settings to ensure that the servo's response to the setpoint has a low error rate.

The following results and discussion pertain to the conditions with a setpoint of 1500 grams, as shown in the graph of the 1500-gram setpoint response for Trial 1 below.



Picture 24: Setpoint response graph, 1500 trials, Trial 1

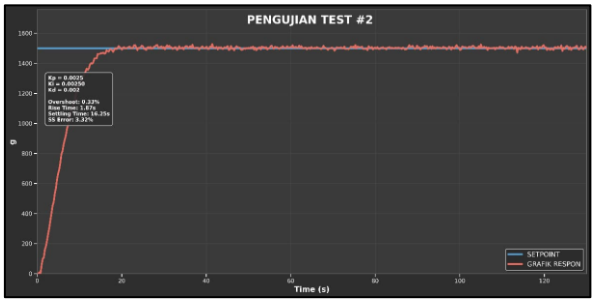
The graph also includes performance metrics:

Table 6: System Performance: Setpoint 1500 Trial 1

Indicator	Result
Setpoint	1500
Ku	0.005
Pu	12
Kp	0.0030
Ki	0.00050
Kd	0.002
Overshoot	4.29%
Rise Time	1.86 s
Settling Time	18.12s
Steady-State Error	4.29%

Based on the PID tuning results table above, the system's performance still requires improvement. With a setpoint of 1500 and a Ku value of 0.005, the applied PID parameters result in an overshoot of 4.29% and a steady-state error of 4.29%, indicating a discrepancy between the system's response and the desired reference value. A rise time of 1.86 seconds is quite good for reaching the target value, but a settling time of 18.12 seconds indicates that the system takes a relatively long time to reach a stable condition. For further PID tuning improvements, the author reduced the Kp value from 0.0030 to approximately 0.0025–0.0028 to reduce the overshoot, as the 4.29% overshoot remains above the expected tolerance in most industrial applications. Second, the author increased the Ki value from 0.00050 to 0.00200–0.00250 to minimize the steady-state error, which currently remains at 4.29%, so that the system can more accurately reach the setpoint value. Third, the author kept the Kd value at 0.002 to adjust the damping and settling time.

Next, testing was conducted in Trial 2 with the setpoint still at 1500, as shown in the following graph of the response to a setpoint of 1500 in Trial 2:



Picture 25: Setpoint response graph: 1,500 trials, trial 2

The graph also includes performance metrics:

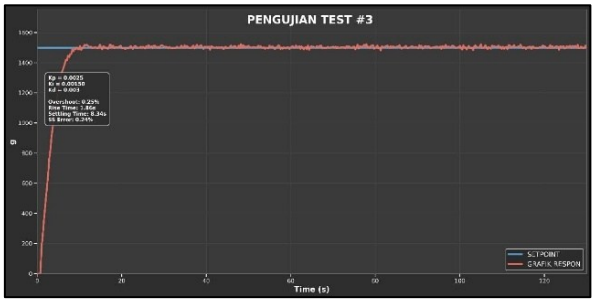
Table 7: System Performance: Setpoint 1500 Trial 2

Indicator	Result
Setpoint	1500
Ku	0,00417
Pu	2
Kp	0,0025
Ki	0.00250
Kd	0.002
Overshoot	0.33%
Rise Time	1.87 s
Settling Time	16.25s
Steady-State Error	3.32%

The results of the second PID tuning iteration show a significant improvement compared to the previous tuning. With a setpoint of 1500 and a Ku value of 0.00417, the new PID parameters have successfully reduced overshoot drastically to just 0.33%, far better than the previous value of 4.29%. The rise time remains consistent at 1.87 seconds, indicating that the system's response to input changes remains just as fast. However, the settling time has improved to 16.25 seconds from the previous 18.12 seconds, indicating that the system reaches a steady-state condition more quickly, although further refinement is still needed. The steady-state error, which remains at 3.32%, indicates that the system is not yet fully accurate in reaching the expected reference value. To achieve more optimal performance in the next tuning, the author maintains the Kp value at 0.0025 as it has proven effective in reducing overshoot to an acceptable level. Second, the author reduced the Ki value from 0.00250 to 0.0020–0.0010 to minimize the steady-state error, which remains at 3.32%, as a stronger integral action will help the system reach the

setpoint with higher precision. Third, the author attempted to slightly increase the Kd value from 0.002 to 0.0025–0.003 to see if the settling time could be further reduced without increasing overshoot.

Next, testing was conducted in Trial 3 with the setpoint still at 1500, as shown in the following graph of the response to a setpoint of 1500 in Trial 3:



Picture 26: Setpoint response graph: 1,500 trials, trial 3

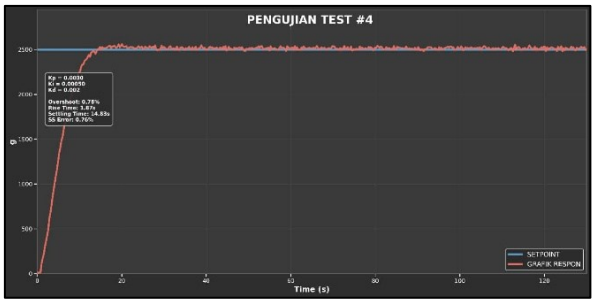
The graph also includes performance metrics:

Table 8: System Performance: Setpoint 1500 Trial 3

Indicator	Result
Setpoint	1500
Ku	0,00417
Pu	3.33
Kp	0,0025
Ki	0.0015
Kd	0.003
Overshoot	0.25%
Rise Time	1.86 s
Settling Time	8.34s
Steady-State Error	0.24%

The results of the third PID tuning iteration show highly satisfactory performance and have met the criteria for optimal system response. With a setpoint of 1500 and a Ku value of 0.00417, the optimized PID parameters produce minimal overshoot of only 0.25%, significantly better than the previous iteration, which reached 0.33%. Most significantly, the settling time was successfully reduced to 8.34 seconds from the previous 16.25 seconds, indicating that the system can reach a stable and steady state much more quickly. The steady-state error was successfully reduced to just 0.24%, which is a remarkable improvement in accuracy compared to the previous iteration, which was still at 3.32%. The rise time, which remains consistent at 1.86 seconds, proves that the system's responsiveness is well maintained.

Next, testing was conducted in Trial 4 with a setpoint of 2500, as shown in the following graph of the response to the 2500 setpoint in Trial 4:



Picture 27: Setpoint response graph: 2,500 trials, trial 4

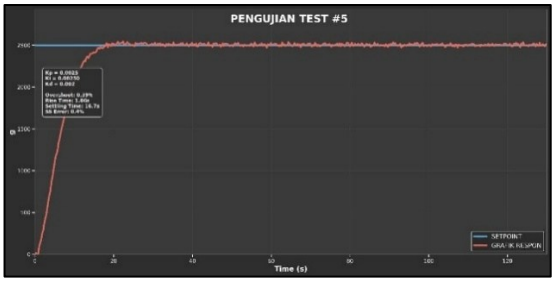
The graph also includes performance metrics:

Table 9: System Performance: Setpoint 1500 Trial 4

Indicator	Result
Setpoint	2500
Ku	0,005
Pu	12
Kp	0.0030
Ki	0.00050
Kd	0.002
Overshoot	0.78%
Rise Time	1.87 s
Settling Time	14.83s
Steady-State Error	0.76%

PID test trial 4 with a setpoint of 2500, using parameters Kp = 0.0030, Ki = 0.00050, and Kd = 0.002, showed a significant decline in performance compared to previous expectations. Overshoot increased to 0.78 % from the previous 0.25%, steady-state error also increased to 0.76% from 0.24%, and settling time, which is a critical parameter, increased drastically to 14.83 seconds from the previous 8.34 seconds. Rise time remained relatively consistent at 1.87 seconds, indicating that the system's initial responsiveness is still well maintained.

Based on the results of Trial 4, the author attempted to use new tuning parameters: Kp = 0.0025, Ki = 0.0025, and Kd = 0.002. Subsequently, testing was conducted in Trial 5 with the setpoint still set at 2500, as shown in the following graph of the response to the 2500 setpoint in Trial 5.



Picture 28: Setpoint response graph: 2,500 trials, trial 5

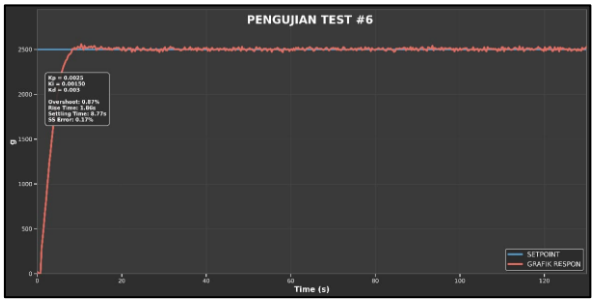
The graph also includes performance metrics:

Table 10: System Performance: Setpoint 2500 Trial 5

Indicator	Result
Setpoint	2500
Ku	0,00417
Pu	2
Kp	0,0025
Ki	0.0025
Kd	0.002
Overshoot	0.39%
Rise Time	1.86 s
Settling Time	16.7s
Steady-State Error	0.4%

PID test trial 5 with a setpoint of 2500, using slightly modified parameters from the previous optimal tuning—namely Kp = 0.0025, Ki = 0.0025, and Kd = 0.002—showed performance that fell between the results of the less-than-satisfactory trial 4. Overshoot increased to 0.39% compared to 0.78% in the previous tuning, steady-state error also increased to 0.4% from the previous 0.76%, and settling time stabilized at 16.7 seconds. Rise time remained consistent at 1.86 seconds, indicating that the system's initial responsiveness was maintained.

Next, testing was conducted in trial 6 with the setpoint still at 2500, as shown in the following graph of the response to the 2500 setpoint in trial 6.



Picture 29: Setpoint response graph: 2,500 trials, trial 6

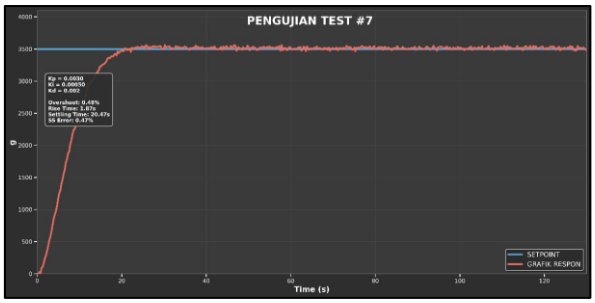
The graph also includes performance metrics:

Table 11: System Performance: Setpoint 2500 Trial 6

Indikator	Nilai
Setpoint	2500
Ku	0,00417
Pu	2
Kp	0,0025
Ki	0.0025
Kd	0.002
Overshoot	0.39%
Rise Time	1.86 s
Settling Time	16.7s
Steady-State Error	0.4%

PID test trial 6 with a setpoint of 2500, using the previously determined optimal tuning parameters namely Kp = 0.0025, Ki = 0.0015, and Kd = 0.003 yielded highly satisfactory results that exceeded previous expectations. Although overshoot increased slightly to 0.87% compared to 0.25% in the 1500 setpoint test, this increase remains within an acceptable range for industrial applications. The steady-state error was successfully reduced to just 0.17%. The settling time of 8.77 seconds indicates that the system can reach a stable condition quickly, with a consistent rise time of 1.86 seconds. The results of Trial 6 confirm that the tuned parameters Kp = 0.0025, Ki = 0.0015, and Kd = 0.003 are the most optimal and reliable final parameters for implementing the control system at various setpoint levels.

Next, testing was conducted for Test 7 with a setpoint of 3500, as shown in the following graph of the response to the 3500 setpoint in Trial 7.



Picture 30: Setpoint 3500 response graph, trial 7

The graph also includes performance metrics:

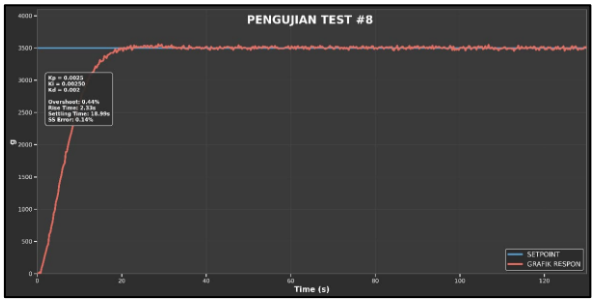
Table 12: System Performance of Setpoint 3500 Trial 7

Indicator	Result
Setpoint	3500
Ku	0,005
Pu	12
Kp	0,003
Ki	0.0005
Kd	0.002
Overshoot	0.48%
Rise Time	1.87 s
Settling Time	20.47s
Steady-State Error	0.47%

Based on the results of Test 7 with a setpoint of 3500, the PID control system demonstrated excellent performance in terms of stability and accuracy. The control parameters used Kp = 0.003, Ki = 0.0005, and Kd = 0.002 resulted in a very small overshoot of 0.48%, indicating that the system is capable of reaching the target without significant value spikes. The steady-state error value of 0.47% also indicates that the system has high accuracy in maintaining the output value close to the desired setpoint. This indicates that the selected combination of control parameters is sufficiently optimal to minimize error and maintain system stability. However, the system's settling time reaches 20.47 seconds. Although the rise time is relatively fast at 1.87 seconds indicating the system can respond well to changes the relatively long settling time suggests that the system requires more time to reach a stable

condition within an acceptable error range.

Next, testing was conducted for Test 8 with the setpoint still at 3500, as shown in the following graph of the response to the 3500 setpoint in Trial 8.



Picture 31: Setpoint 3500 response graph, trial 8

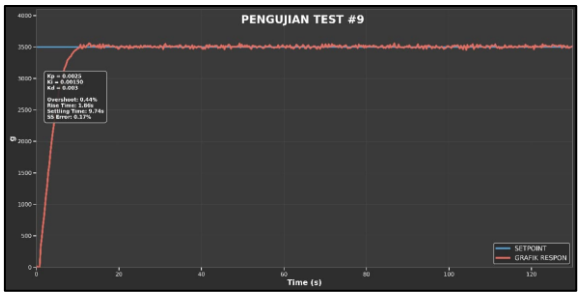
The graph also includes performance metrics:

Table 13: System Performance: Setpoint 3500 Trial 8

Indicator	Result
Setpoint	3500
Ku	0,00417
Pu	2
Kp	0,0025
Ki	0.0025
Kd	0.002
Overshoot	0.44%
Rise Time	2.33 s
Settling Time	18.99s
Steady-State Error	0.14%

Based on the results of trial 8 with a setpoint of 3500, the PID control system showed a significant improvement in performance compared to trial 7. The control parameters used were Kp = 0.0025, Ki = 0.0025, and Kd = 0.002, with Ku = 0.00417 and Pu = 2, resulting in a much smaller steady-state error of 0.14%, down from 0.47% in Trial 7. This reduction in error indicates that the system has higher accuracy in maintaining the output value at the desired setpoint. The resulting overshoot also improved slightly to 0.44%, lower than the 0.48% in the previous trial, indicating that the system is becoming more stable with minimal value spikes upon reaching the target. In terms of response time, Trial 8 showed a significant improvement, particularly in settling time, which decreased to 18.99 seconds from the previous 20.47 seconds. Although rise time increased slightly to 2.33 seconds from the previous 1.87 seconds, this trade-off is still acceptable given the significant improvement in system accuracy demonstrated by the much lower steady-state error.

Next, testing was conducted for Test 9 with the setpoint still at 3500, as shown in the following graph of the response to the 3500 setpoint for Trial 9.



Picture 32: Setpoint 3500 response graph, trial 9

The graph also includes performance metrics:

Table 14: System Performance: Setpoint 3500 Trial 9

Indicator	Result
Indikator	Nilai
Setpoint	3500
Ku	0,00417
Pu	3.33
Kp	0,0025
Ki	0.0015
Kd	0.003
Overshoot	0.44%
Rise Time	1.86 s
Settling Time	9.74s

Based on the results of Trial 9 with a setpoint of 3500, the PID control system demonstrated the best performance among the three trials conducted. The control parameters used were Kp = 0.0025, Ki = 0.0015, and Kd = 0.003, with values of Ku = 0.00417 and Pu = 3.33, resulting in a highly optimal performance combination. The settling time decreased drastically to 9.74 seconds. This significant improvement indicates that the system is capable of reaching a stable condition in a shorter time. The rise time also showed the best result at 1.86 seconds, indicating that the system has a fast yet stable response.

Table 15:PID Test Data

NO	Ku	Pu (s)	KP	KI	KD	OVERSHOOT (%)	RISE TIME (S)	SETTLING TIME(S)	STEADY STATE ERROR (%)	SET POINT (G)	FINAL_WEIGHT (G)
1	0.005	12	0.003	0.0005	0.002	4.29	1.86	18.12	4.29	1500	1542.86
2	0.00417	2	0.0025	0.0025	0.002	0.33	1.87	16.25	3.32	1500	1503.32
3	0.00417	3.33	0.0025	0.0015	0.003	0.25	1.86	8.34	0.24	1500	1502.38
4	0.005	12	0.003	0.0005	0.002	0.78	1.87	14.83	0.76	2500	2515
5	0.00417	2	0.0025	0.0025	0.002	0.39	1.86	16.7	0.4	2500	2507.93

6	0.00417	3.33	0.0025	0.0015	0.003	0.87	1.86	8.77	0.17	2500	2503.33
7	0.005	12	0.003	0.0005	0.002	0.48	1.87	20.47	0.47	3500	3514.21
8	0.00417	2	0.0025	0.0025	0.002	0.44	2.33	18.99	0.14	3500	3495
9	0.00417	3.33	0.0025	0.0015	0.003	0.44	1.86	9.74	0.17	3500	3505.25

Based on the results of testing several combinations of PID parameters in the PID test table, it can be concluded that changes in the values of Kp, Ki, and Kd significantly affect system performance, particularly in terms of overshoot, settling time, and steady-state error. Among all the tests conducted, the PID parameter combination with Kp = 0.00417, Ki = 0.0025, and Kd = 0.0015 demonstrated the most optimal performance, characterized by low overshoot, relatively fast settling time, and low steady-state error.

Furthermore, the system was able to reach the setpoint effectively under various load variations (1500 g, 2500 g, and 3500 g), indicating that the applied PID control is sufficiently stable and responsive. However, since the PID tuning process was still performed using trial and error, the system's performance has the potential to be improved through automatic tuning methods to ensure more consistent system response to every load change.

4. Conclusion

Based on the testing and analysis conducted, this Final Project yields several conclusions formulated based on the analysis of sensor calibration test results, system performance and integration with the PLC, GUI, and SQL Server database, as well as PID control of the servo motor. Thus, the conclusions outlined in this chapter are expected to address the research questions and demonstrate that the research objectives have been achieved as planned. The conclusions that can be drawn are as follows:

1. Load Cell Calibration and Accuracy Testing

The results of the load cell calibration and accuracy tests indicate an excellent level of accuracy, with reading deviations below ±0.2% relative to the digital scale used as a reference. These values remain within the standard industry tolerance limits of PT Simatelex Manufactory Batam; therefore, the load cell is deemed suitable and reliable for use as the primary instrument in measuring tensile force on the Connector Wire Testing Machine.

2. Testing the Latency and Responsiveness of ESP32–GUI Communication

Based on the test results, the integration between the ESP32 and the Visual C# GUI was found to be responsive and reliable. An average delay of 2.2 ms demonstrates that the system operates well below the standard maximum limit of <100 ms. This responsiveness indicates that the communication mechanisms, data transmission processes, and message handling on both sides are functioning optimally.

3. Data Processing Performance Testing on a Database

Based on the results of system testing on 150 data samples, a success rate of 100%, a duplicate rate of 0%, and a coefficient of variation (CV) of 11% were obtained. These results indicate that the data acquisition and storage processes performed very well without any data loss or duplication, and that the round-trip time (RTT) stability level falls within the stable category.

4. PID Performance Testing for Cable Tension Stabilization

Test results of the PID controller on the system indicate that the application of proportional, integral, and derivative control improves the stability and dynamic response of the servo motor during cable tensile strength testing. Without PID control, the motor tends to experience overshoot, speed fluctuations, and a relatively longer response time to reach a stable condition. This proves that the implementation of a PID controller not only improves the accuracy of motor control but also enhances the consistency of test results and the overall reliability of the system.

5. Bibliography

[1] D.C. Permana, R. Ferdiansyah, F. P. Safira, Z. T. A. Gumilang, A. J. Pangestu, and R. W. Abdul Rozak, "Industrial Automation: An Opportunity or a Threat," J. Community Service. Empowerment, Innovation, and Change, vol. 3, no. 3, pp. 139–146, 2023, doi: 10.59818/jpm.v3i3.515.

[2] H. P. Batam, "Working Visit by the Board of Directors of PT SIER to PT Persero Batam," perserobatam. [Online]. Available: https://perserobatam.com/kunjungan-kerja-direksi-pt-sier-ke-pt persero-batam/

[3] Superadmin, "Conveyor Fabrication Services in Batam," PT. Multi Karya Teknik. [Online]. Available: https://multikaryatehnik.co.id/jasa-fabrikasi- conveyor-di-batam/

[4] FIrawan, A. B. Neris, R. A. Marlina, T. Pertambangan, and S. Tinggi, Padang Institute of Technology, "Analysis of Belt Conveyor Productivity in the Main Shaft Tunnel at PT Allied Indo Coal Jaya (AICJ) in Parambahan, Talawi District, Sawahlunto City, West Sumatra," Journal of Science and Technology, 2020.

[5] B. Dhiya' Ushofa, L. Anifah, G. Buditjahjanto, and Endryansyah, "Speed Control System for a DC Motor in a Conveyor Using the PID Control Method," J. Electrical Engineering, vol. 11, no. State University of Surabaya, pp. 332–342, 2022.

[6] W. WALUYO, A. FITRIANSYAH, and S. SYAHRIAL, "Analisis Penalaan Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban menggunakan Metode Heuristik," ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron., vol. 1, no. 2, p. 79, 2013, doi: 10.26760/elkomika.v1i2.79.

[7] A. Pane, J., Surya, A., Novita, S., Mazmur, R., Aryza, A., Hamdani, Rizky, "Implementation of PID in Motor Control Using the PID Method and the Atmega Microcontroller," Sainteks, vol. 1, no. 1, pp. 196–201, 2019, [Online]. Available:https://seminar.id.com/prosiding/index.php/sainteks/article/download/155/153

[8] K. P. Mentor, "Design of a DC Motor Speed Control System Using a Proportional-Integral-Derivative Controller for a Parking Gate," J. Electrical Eng., vol. 12, p. 48, 2023.

[9] M. M. I. Putra, S. R. U. A. Sompie, and S. Paturusi, "Implementation of Speech Recognition in an English Language Learning Application for Children," J. Inform. Tech., vol. 15, no. 4, pp. 247–256, 2020, [Online] Available: https://ejournal.unsrat.ac.id/index.php/informatika/article/view/30426

[10] S. STEKOM, "C# (programming language)," Encyclopedia. Accessed: Mar. 12, 2024 Available:https://p2k.stekom.ac.id/ensiklopedia/C_Sharp_(bahasa_pemrograman)

[11] M. R. A. Nurkholis Putera and R. Hidayat, "Speed Control of a DC Motor Using a PID Controller with an Encoder as Feedback," STRING (Journal of Research and Innovation Technology, vol. 7, no. 1, p. 50, 2022, doi: 10.30998/string.v7i1.13026.

[12] R. Birdayansyah, N. Sudjarwanto, and O. Zebua, "Speed Control of a DC Motor Using Voice Commands Based on an Arduino Microcontroller," Electr. – J. of Electrical Engineering and Technology, vol. 9, no. 2, pp. 97–107, 2015.

[13] M. A. Ulum and S. I. Haryudo, "Design of an Internet of Things-Based DC Motor Rotational Speed Monitoring System Using the BLYNK Application," J. Electrical Eng., vol. 9, no. 1, pp. 855–862, 2020.

[14] Honorine Angue Mintsa, G. E. (2023). Optimal Tuning of PID Controller Gains Using the Ziegler-Nichols Approach for an Electrohydraulic Servo System. Volume 25, Issue 11, Pages 158–166, 2023; Article No. JERR.110033, 25, 159–166.

[15] Muhammad Nabel Al Fayyed, R. M. (2025). Optimization of BLDC PID Control Using the Ziegler-Nichols Method. JURITEK: Scientific Journal of Mechanical, Electrical, and Computer Engineering 2025 (July), vol. 5, no. 2, Nabel Al Fayyed, et al., 5.

[16] Syamsudduha Syahrorini, M. A. (2025). Optimasi Pengendalian PID Motor DC Menggunakan Metode Ziegler–Nichols dengan Encoder dan Op-Amp (LM324). Vol. 10 No. 2 (2025): 10 Desember, hlm. 7–15.

[17] Wicaksono, H. (2004). Analysis of the Performance and Robustness of Several PID Controller Tuning Methods for DC Motors. Journal of Electrical Engineering, Vol. 4, No. 2, September 2004: 70–78, 4, 70–78.

[18] Mila Diah Ika Putri, A. M. (2022). PENGENDALI KECEPATAN SUDUT MOTOR DC MENGGUNAKAN KONTROL PID DAN TUNING ZIEGLER NICHOLS. Vol.23, No.1, April 2022, Hal. 09-18, 23, 09-18.

[19] Refigol Afrawira, R. F. (2023). Comparative Analysis of PID Controllers for DC Motors Using the Ziegler-Nichols Method and Trial-and-Error Method. Received: April 5, 2021, Revised: May 1, 2023, Published: May 22, 2023, 5, 210–218.

[20] Yash Mewada, S. P. (2022). Sistem Pengendalian PID dan Penyetelan Otomatis PID Menggunakan Metode Ziegler-Nichols. International Journal Publishing INFLUENCE: Jurnal Internasional Ulasan Ilmiah Volume 4, No. 1, 2022, hlm. 249–253.

[21] Dessy Dwi Anggraini, S. P. (2025). PID - Metode Ziegler Nichols untuk Pengendalian Frekuensi Beban. Jurnal Teknologi Terkini dan Multidisiplin, 4, 63-67.