

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE  
MINAS GERAIS - *CAMPUS* SABARÁ  
BACHARELADO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Geraldo Junio Nelcino da Silva

**DESENVOLVIMENTO DE UMA PLATAFORMA DIDÁTICA PARA  
CONTROLE E SUPERVISÃO DE UM SISTEMA DE TANQUES**

Sabará  
2026

GERALDO JUNIO NELCINO DA SILVA

**DESENVOLVIMENTO DE UMA PLATAFORMA DIDÁTICA PARA  
CONTROLE E SUPERVISÃO DE UM SISTEMA DE TANQUES**

Trabalho de Conclusão de Curso apresentado à banca examinadora do curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais *Campus* Sabará, como parte dos requisitos para obtenção do título de Bacharel em Engenharia de Controle e Automação.

**Orientador:** Profa. Dra. Mariella Maia Quadros

**Coorientador:** Prof. Dr. Rodrigo Hiroshi Murofushi

Sabará  
2026

Silva, Geraldo Junio Nelcino da

S586d

Desenvolvimento de uma plataforma didática para controle e supervisão de um sistema de tanques [manuscrito]. / Geraldo Junio Nelcino da Silva. - 2026.

84 f. : il.

Orientação: Profa. Dra. Mariella Maia Quadros.

Coorientadora: Prof. Dr. Rodrigo Hiroshi Murofushi.

Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Controle e Automação) – Instituto Federal de Minas Gerais, *Campus* Sabará.

1. Engenharia – Plataforma didática. – Monografia. 2. Controle de processo. – Monografia. 3. Controladores programáveis. – Monografia. 4. Controladores PID. – Monografia. 5. Sistemas de controle supervisão. – Monografia. 6. Controle automático. – Monografia. I. Murofushi, Rodrigo Hiroshi. II. Murofushi, Rodrigo Hiroshi. III. Instituto Federal de Minas Gerais, *Campus* Sabará. IV. Bacharelado em Engenharia de Controle e Automação. V. Título.

CDU 681.5

César dos Santos Moreira / CRB6-2229  
Biblioteca do IFMG *Campus* Sabará



**MINISTÉRIO DA EDUCAÇÃO**  
**SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA**  
**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS GERAIS**  
**Campus Sabará**

**Diretoria de Ensino, Pesquisa e Extensão**  
**Coordenação do Curso de Engenharia de Controle e Automação**  
Rodovia MGC 262, Km 10 - Bairro Sobradinho - CEP 34590-390 - Sabará - MG - www.ifmg.edu.br

## **ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO**

Aos quatro dias do mês de fevereiro do ano de dois mil e vinte e seis, às dezenove horas, sob a presidência da professora Mariella Maia Quadros, reuniu-se a banca examinadora composta pelos professores abaixo relacionados, para a defesa do Trabalho de Conclusão de Curso (TCC) do discente **Geraldo Junio Nelcino da Silva**, matrícula nº 0065764, do curso de Engenharia de Controle e Automação, do IFMG *campus* Sabará.

O trabalho intitulado **DESENVOLVIMENTO DE UMA PLATAFORMA DIDÁTICA PARA CONTROLE E SUPERVISÃO DE UM SISTEMA DE TANQUES** foi apresentado e submetido à apreciação da banca. Após exposição, arguição e deliberação, a banca atribuiu a nota final de 80 pontos (de um total de 80). Somados aos 18 pontos atribuídos ao aluno pelo docente responsável pela disciplina de Projeto II, o aluno ficou com um total de 98 pontos, resultando em **aprovado**, condicionado ao cumprimento das orientações e prazos estabelecidos pelas normas acadêmicas institucionais.

Compuseram a Banca Examinadora:

- Membro 1 (Presidente): Profa. Mariella Maia Quadros (orientadora)
- Membro 2: Prof. Maicon Vaz Moreira
- Membro 3: Prof. Vinicius Martins Almeida

O discente deverá apresentar a versão final do trabalho em formato PDF e depositá-la no repositório institucional até o dia vinte e sete de fevereiro de dois mil e vinte e seis. O não cumprimento dessas exigências implicará na não contabilização das horas referentes aos componentes curriculares de TCC I e TCC II no sistema acadêmico.

A sessão foi encerrada às vinte horas e trinta minutos. Para constar, eu, Rodrigo Mariella Maia Quadros, redigi a presente ata que após lida publicamente, foi aprovada e assinada pelos membros da banca examinadora.

Sabará, 05 de fevereiro de 2026.



Documento assinado eletronicamente por **Mariella Maia Quadros, Professora EBTT**, em 05/02/2026, às 20:08, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Vinicius Martins Almeida, Professor EBTT**, em 05/02/2026, às 20:24, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Maicon Vaz Moreira, Professor EBTT**, em 06/02/2026, às 08:06, conforme Decreto nº 10.543, de 13 de novembro de 2020.



A autenticidade do documento pode ser conferida no site <https://sei.ifmg.edu.br/consultadocs> informando o código verificador **2612413** e o código CRC **8AA1F4CC**.

23714.001466/2025-48	2600082v1
----------------------	-----------

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, pela força, sabedoria e proteção concedidas ao longo de toda esta caminhada, tornando possível a conclusão de mais esta etapa. Expresso minha profunda gratidão aos meus pais, Glória de Fátima Silva e Adair Nelcino da Silva, pelo amor, apoio incondicional, incentivo constante e compreensão nos momentos mais desafiadores da minha formação acadêmica.

Agradeço de forma especial à minha orientadora Mariella Maia Quadros, pela orientação técnica, dedicação e disponibilidade ao longo do desenvolvimento deste trabalho, contribuindo de maneira fundamental para sua realização. Estendo meus agradecimentos ao meu coorientador Rodrigo Hiroshi Murofushi, pelas valiosas contribuições, sugestões e apoio técnico durante o projeto.

Registro também meu agradecimento ao meu amigo Guilherme Gomes, pelo apoio, colaboração e pela disponibilização de dispositivos essenciais para a montagem prática da plataforma experimental. Por fim, agradeço a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho.

“A ciência pode explicar o que existe;  
a engenharia cria o que nunca existiu.”

Theodore von Kármán

## RESUMO

Este trabalho apresenta o desenvolvimento de uma plataforma didática para o controle e supervisão de um sistema de tanques acoplados, visando auxiliar o ensino prático na disciplina de Controle de Processos Industriais do curso de Engenharia de Controle e Automação do IFMG – campus Sabará. A solução proposta integra programação em Controlador Lógico Programável (CLP) e sistema de supervisão via *software* BluePlant, permitindo a implementação e análise de controladores PID (Proporcional-Integral-Derivativo), bem como de compensadores de avanço e de atraso de fase. A metodologia contempla a análise e modelagem do sistema em malha aberta, a partir da aplicação de degraus de diferentes amplitudes e pontos de operação, possibilitando a obtenção da dinâmica do processo. Também inclui a elaboração e discretização de controladores pelo método de Tustin, além da execução de experimentos em malha fechada com análise gráfica dos resultados. Espera-se que a plataforma contribua para o aprendizado aplicado dos estudantes, proporcionando maior interação com sistemas reais e comparações diretas entre diferentes estratégias de controle, alinhando a formação acadêmica às demandas da Indústria 4.0.

**Palavras-chave:** Controle de processo; CLP; Sistemas Supervisórios; PID; Compensadores clássicos.

## ABSTRACT

This work presents the development of a didactic platform for the control and supervision of a coupled-tank system, aiming to support practical learning in the Industrial Process Control course of the Control and Automation Engineering program at IFMG – Sabará campus. The proposed solution integrates Programmable Logic Controller (PLC) programming and supervision through the BluePlant software, enabling the implementation and analysis of PID (Proportional-Integral-Derivative) controllers as well as lead and lag compensators. The methodology includes the analysis and modeling of the system in open loop by applying step inputs with different amplitudes and operating points, allowing the identification of the process dynamics. It also covers the design and discretization of controllers using the Tustin method, followed by closed-loop testing with graphical response analysis. The platform is expected to enhance students' applied learning experience by providing greater interaction with real systems and direct comparisons between different control strategies, aligning academic training with Industry 4.0 requirements.

**Keywords:** Process control; PLC; Supervisory systems; PID; Classic compensators.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Bancada de ensaios. . . . .	15
Figura 2 – Diagrama do sistema idealizado. . . . .	19
Figura 3 – Bancada didática para controle de nível e temperatura. . . . .	20
Figura 4 – Tela de supervisão do controlador PID. . . . .	21
Figura 5 – Sistema de malha fechada. . . . .	23
Figura 6 – Diagrama esquemático da bancada de ensaios. . . . .	33
Figura 7 – Transmissor de pressão piezoresistivo. . . . .	34
Figura 8 – Bomba de recirculação. . . . .	34
Figura 9 – Válvula solenoide. . . . .	35
Figura 10 – Curva de ajuste. . . . .	40
Figura 11 – Tela Inicial do Sistema. . . . .	42
Figura 12 – Tela de Resposta ao Degrau em Malha Aberta. . . . .	43
Figura 13 – Tela de Resposta ao Controlador de Avanço de Fase. . . . .	43
Figura 14 – Tela de Resposta ao Controlador de Atraso de Fase. . . . .	44
Figura 15 – Tela do Controlador PID. . . . .	45
Figura 16 – Configuração da plataforma na bancada de ensaios. . . . .	47
Figura 17 – Gráfico da resposta em malha aberta do sistema apresentada pelo supervisor. . . . .	48
Figura 18 – Degrau em Malha Aberta no Scilab. . . . .	49
Figura 19 – Degrau em Malha Aberta com a linha de resposta da função. . . . .	50
Figura 20 – Resposta ao degrau da simulação do sistema com o controlador de avanço de fase. . . . .	52
Figura 21 – Tela do supervisor com parâmetros inseridos e resposta prática do sistema. . . . .	53
Figura 22 – Resposta ao degrau obtida no Scilab a partir dos dados exportados. . . . .	54
Figura 23 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador de Avanço de Fase. . . . .	55
Figura 24 – Resposta ao degrau da simulação do sistema com o controlador PI. . . . .	57
Figura 25 – Tela do supervisor com parâmetros inseridos e resposta prática do sistema com o controlador PI. . . . .	58
Figura 26 – Resposta ao degrau obtida no Scilab a partir dos dados exportados do controlador PI. . . . .	59
Figura 27 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador PI. . . . .	59
Figura 28 – Resposta a simulação do sistema à entrada em rampa. . . . .	63
Figura 29 – Resposta ao degrau da simulação do sistema com o controlador de Atraso de Fase. . . . .	64
Figura 30 – Tela do supervisor com parâmetros inseridos e resposta prática do sistema com o controlador de Atraso de Fase. . . . .	65

Figura 31 – Resposta ao degrau obtida no Scilab a partir dos dados exportados do controlador de Atraso de Fase. . . . .	66
Figura 32 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador Atraso de Fase. . . . .	67

## **LISTA DE TABELAS**

Tabela 1 – Valores de calibração entre a leitura do CLP e o nível em milímetros. . . . .	39
Tabela 2 – Lista de <i>tags</i> do supervisor e seus respectivos endereços Modbus . . . . .	46

## LISTA DE ABREVIATURAS E SIGLAS

CLP	Controlador Lógico Programável
PID	Proporcional-Integral-Derivativo
PI	Proporcional-Integral
IHM	Interface Homem-Máquina
SCADA	<i>Supervisory Control and Data Acquisition</i>
ST	<i>Structured Text</i> (Texto Estruturado)
IEC	<i>International Electrotechnical Commission</i>
TCP/IP	<i>Transmission Control Protocol / Internet Protocol</i>
OPC	<i>OLE (Object Linking and Embedding) for Process Control</i>
PWM	<i>Pulse Width Modulation</i> (Modulação por Largura de Pulso)
LIT	Lineares Invariantes no Tempo
IFMG	Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
<b>1.1</b>	<b>Objetivos</b>	<b>16</b>
<b>1.2</b>	<b>Justificativa</b>	<b>16</b>
<b>1.3</b>	<b>Organização do Texto</b>	<b>18</b>
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	<b>19</b>
<b>3</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>22</b>
<b>3.1</b>	<b>Sistemas de controle</b>	<b>22</b>
<i>3.1.1</i>	<i>Sistemas de segunda ordem e parâmetros de desempenho</i>	<i>23</i>
<b>3.2</b>	<b>Modelagem Matemática de Sistemas</b>	<b>24</b>
<b>3.3</b>	<b>Controlador PID</b>	<b>24</b>
<b>3.4</b>	<b>Compensadores de Avanço de Fase e Atraso de Fase</b>	<b>25</b>
<i>3.4.1</i>	<i>Compensador de Avanço de Fase</i>	<i>26</i>
<i>3.4.2</i>	<i>Compensador de Atraso de Fase</i>	<i>27</i>
<b>3.5</b>	<b>Sistemas no domínio do tempo contínuo e discreto</b>	<b>27</b>
<i>3.5.1</i>	<i>Discretização pelo Método de Tustin</i>	<i>28</i>
<b>3.6</b>	<b>Sistemas Supervisórios</b>	<b>29</b>
<i>3.6.1</i>	<i>Comunicação e Protocolos de Rede</i>	<i>30</i>
<i>3.6.2</i>	<i>Tags</i>	<i>31</i>
<b>3.7</b>	<b>Linguagem de Programação em Texto Estruturado</b>	<b>32</b>
<b>4</b>	<b>METODOLOGIA</b>	<b>33</b>
<b>5</b>	<b>DESENVOLVIMENTO E RESULTADOS</b>	<b>37</b>
<b>5.1</b>	<b>Elaboração das Equações de Diferenças</b>	<b>37</b>
<i>5.1.1</i>	<i>Controlador PID Discreto no Tempo</i>	<i>37</i>
<i>5.1.2</i>	<i>Compensadores de Avanço de Fase e Atraso de Fase Discretos no Tempo</i>	<i>37</i>
<b>5.2</b>	<b>Calibração do Sensor de Nível</b>	<b>38</b>
<b>5.3</b>	<b>Implementação do Código e Configuração da Saída de Controle</b>	<b>40</b>
<b>5.4</b>	<b>Desenvolvimento das Telas do Sistema Supervisório</b>	<b>41</b>
<i>5.4.1</i>	<i>Tela Inicial do Sistema</i>	<i>41</i>
<i>5.4.2</i>	<i>Tela de Resposta ao Degrau em Malha Aberta</i>	<i>42</i>

5.4.3	<i>Tela do Controlador de Avanço de Fase</i> . . . . .	43
5.4.4	<i>Tela do Controlador de Atraso de Fase</i> . . . . .	44
5.4.5	<i>Tela do Controlador PID</i> . . . . .	44
5.5	<b>Comunicação do Supervisório com o CLP</b> . . . . .	45
5.6	<b>Configuração da Plataforma na Bancada de Ensaios</b> . . . . .	47
5.7	<b>Obtenção da Resposta em Malha Aberta</b> . . . . .	48
5.8	<b>Controlador de Avanço de Fase</b> . . . . .	51
5.8.1	<i>Cálculo dos Parâmetros do Controlador</i> . . . . .	51
5.8.2	<i>Simulação no Scilab</i> . . . . .	52
5.8.3	<i>Resultado Prático no Supervisório</i> . . . . .	53
5.9	<b>Controlador PI (implementação a partir do PID)</b> . . . . .	55
5.9.1	<i>Cálculo dos Parâmetros do Controlador</i> . . . . .	56
5.9.2	<i>Simulação no Scilab</i> . . . . .	57
5.9.3	<i>Implementação e Resultado Prático no Supervisório</i> . . . . .	58
5.10	<b>Controlador de Atraso de Fase</b> . . . . .	60
5.10.1	<i>Cálculo dos Parâmetros do Controlador</i> . . . . .	61
5.10.2	<i>Simulação no Scilab</i> . . . . .	63
5.10.3	<i>Implementação e Resultado Prático no Supervisório</i> . . . . .	65
6	<b>CONCLUSÃO</b> . . . . .	68
	<b>REFERÊNCIAS</b> . . . . .	69
	<b>APÊNDICE A – CÓDIGO-FONTE DO SISTEMA EM TEXTO ESTRUTURADO</b>	
	<b>(ST)</b> . . . . .	71
A.1	<b>Declaração de Variáveis Globais</b> . . . . .	71
A.2	<b>Declaração de Variáveis Locais</b> . . . . .	72
A.3	<b>Trecho principal do código</b> . . . . .	73

# 1 INTRODUÇÃO

Nos últimos anos, os avanços da automação industrial têm sido amplamente impulsionados pelos conceitos da Indústria 4.0, que incluem a integração de sistemas ciberfísicos, monitoramento e comunicação em tempo real e análise de dados para a realização de controle e otimização de processos de produção (RIBEIRO; ABREU, 2020).

Sistemas de controle de malha fechada são conjuntos de dispositivos e estratégias projetados para gerenciar, comandar e regular o comportamento de diversos processos. Eles operam de forma a monitorar variáveis de saída, aplicando correções nas variáveis de entrada para alcançar o desempenho desejado. Esses sistemas são amplamente utilizados em diversos setores industriais, com o objetivo de garantir maior precisão, estabilidade e confiabilidade nas operações (MAYA; LEONARDI, 2014).

Uma das ferramentas mais importantes utilizadas para o monitoramento de processos são os sistemas supervisórios, que desempenham um papel essencial no monitoramento e controle de processos industriais. Esses sistemas são comumente implementados por meio de arquiteturas SCADA (do inglês, *Supervisory Control and Data Acquisition*), proporcionando aos usuários uma interface para acompanhar o desempenho dos equipamentos e variáveis do processo em tempo real, permitindo que informações cruciais sejam processadas e exibidas de forma acessível e dinâmica (MORAES; CASTRUCCI, 2010).

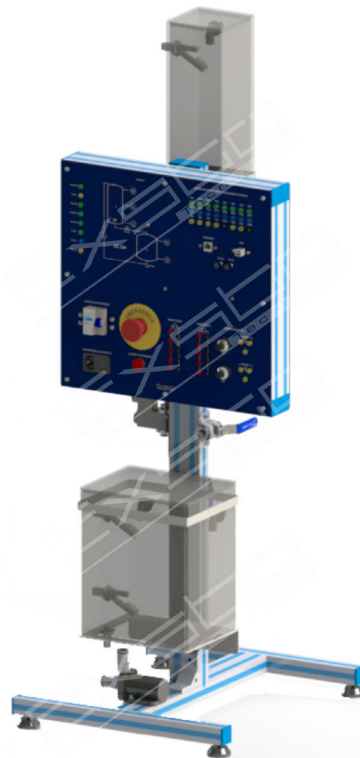
Os sistemas supervisórios são indispensáveis em diversos processos industriais, como, por exemplo, em sistemas de controle de nível de tanques, que desempenham funções cruciais em setores químicos, alimentícios, de tratamento de água, entre outros, sendo utilizados para armazenamento e mistura de reagentes, controle de processos como pasteurização e fermentação, além de garantir a qualidade de líquidos tratados (MORAES; CASTRUCCI, 2010). Eles representam um dos principais exemplos de aplicação de métodos de controle contínuo, sendo frequentemente empregados no controle de variáveis como nível, vazão e pressão (OGATA, 2010).

Em ambientes acadêmicos, bancadas didáticas, como por exemplo, de sistemas de tanques acoplados, são fundamentais para o aprendizado prático dos alunos sobre métodos de controle e análise do comportamento dinâmico dos processos. Conforme especificado no manual do fabricante Exsto Tecnologia Ltda (2021), os sistemas de tanques acoplados consistem, geralmente, em dois ou mais reservatórios interligados por tubulações e válvulas, permitindo o fluxo de fluido entre eles. Esse arranjo permite simular processos industriais reais e integra teoria e prática, permitindo que os estudantes experimentem e validem conceitos fundamentais da área de controle.

No Laboratório de Controle e Processos Industriais do Instituto Federal de Minas Gerais (IFMG) campus Sabará, há uma bancada de ensaios da fabricante Exsto, integrada a um Controlador Lógico Programável (CLP) da Altus, que permite a realização de simulações e implementações de métodos para o controle de nível e vazão. Contudo, na disciplina de Controle de Processos Industriais do curso de Bacharelado em Engenharia de Controle e Automação,

foram observadas dificuldades em obter uma análise gráfica em tempo real do sistema, bem como na elaboração, implementação e avaliação de diferentes tipos de controladores, como o controlador de avanço de fase, o controlador de atraso de fase e o controlador Proporcional-Integral-Derivativo (PID). Essas limitações dificultam a transição dos conceitos teóricos para a prática, comprometendo a preparação dos alunos para os desafios da indústria, em que o domínio do projeto, da implementação e da análise de controladores é fundamental para o desenvolvimento de soluções eficientes e alinhadas às exigências dos processos industriais. A Figura 1 apresenta o modelo de bancada de ensaios disponível no IFMG campus Sabará.

Figura 1 – Bancada de ensaios.



Fonte: (Exsto Tecnologia Ltda, 2021).

Diante desse cenário, este trabalho tem como objetivo desenvolver uma plataforma didática para o controle e supervisão do sistema didático de tanques, presente no laboratório de Controle de Processos Industriais. A plataforma permitirá ao usuário obter a resposta em degrau tanto em malha aberta, para obtenção do modelo matemático do sistema, quanto em malha fechada, para análise da resposta ao controle, além da geração e implementação dos controladores de avanço de fase, atraso de fase e PID. A interface desenvolvida possibilitará ao usuário inserir diretamente os parâmetros dos controladores projetados, facilitando os experimentos e ajustes durante a execução dos algoritmos. Dessa forma, espera-se não apenas auxiliar os alunos na compreensão dos conceitos de controle, mas também prepará-los para enfrentar os desafios industriais, contribuindo para o desenvolvimento de profissionais mais capacitados.

## 1.1 Objetivos

O objetivo geral deste trabalho é desenvolver uma plataforma didática para o controle e supervisão de um sistema didático de tanques, que permita a realização de experimentos em malha aberta para a modelagem do sistema e a implementação, análise e comparação dos desempenhos dos controladores projetados para o sistema em malha fechada, visando facilitar o aprendizado prático dos alunos do curso de Engenharia de Controle e Automação.

Os objetivos específicos são:

- Desenvolver a interface supervisória e os recursos de aquisição de dados para a realização de ensaios de resposta ao degrau em malha aberta, definindo os parâmetros de teste e coletando os dados experimentais para a modelagem matemática do sistema;
- Realizar a modelagem matemática do sistema a partir dos dados obtidos em malha aberta, visando a identificação da dinâmica do processo;
- Implementar, no CLP, os algoritmos de cálculo e discretização dos controladores de avanço de fase, atraso de fase e PID;
- Desenvolver, na interface supervisória, os campos de entrada de dados e as telas necessárias para a configuração dos experimentos dos controladores em malha fechada;
- Permitir a visualização gráfica, em tempo real, da resposta do sistema em malha fechada para os controladores implementados;
- Validar a plataforma desenvolvida por meio da comparação entre os resultados obtidos nos ensaios práticos e as simulações realizadas no Scilab, verificando a coerência entre o modelo matemático e o comportamento do sistema real;
- Analisar e comparar o desempenho dos controladores projetados com base em indicadores clássicos, como tempo de subida, tempo de acomodação, sobressinal e erro em regime permanente.

## 1.2 Justificativa

O avanço tecnológico proporcionado pela Indústria 4.0 tem elevado as exigências sobre a qualificação dos profissionais da área de automação industrial, que precisam não apenas dominar conceitos teóricos, mas também desenvolver habilidades práticas para aplicação desses conhecimentos em ambientes reais (SILVA *et al.*, 2019). Apesar da existência de bancadas didáticas no IFMG – Sabará, observou-se que as limitações nas ferramentas disponíveis para análise e supervisão em tempo real dificultam a consolidação do aprendizado dos alunos na disciplina de Controle de Processos Industriais.

A ausência de uma plataforma integrada que permita a visualização gráfica dinâmica e a comparação de diferentes controladores compromete o desenvolvimento de competências essenciais, como a análise crítica e a tomada de decisão baseada em dados reais do processo.

Outro grande problema enfrentado pelos alunos é que, ao iniciarem a disciplina de Controle de Processos Industriais, os mesmos ainda não tiveram contato aprofundado com linguagens de programação utilizadas na automação, como as linguagens Ladder e texto estruturado, aplicadas em CLPs. Essa falta de conhecimento das linguagens de programação acaba gerando ainda mais dificuldades no aprendizado prático, já que os alunos precisam não apenas compreender os conceitos de controle, mas também desenvolver e implementar os códigos dos controladores projetados, de forma a executar os comandos e operações necessários para que os comportamentos e resultados desejados sejam alcançados.

Diante disso, o desenvolvimento de uma plataforma didática que una programação e supervisão de controladores representa uma resposta direta a essa lacuna educacional, contribuindo para a formação de engenheiros mais preparados para os desafios atuais da indústria, oferecendo um ambiente mais acessível e interativo para os estudantes, permitindo que foquem na compreensão do comportamento dos sistemas de controle e nos resultados dos controladores aplicados. O uso de *softwares* amplamente adotados no mercado, como o MasterTool e o BluePlant na construção dessa plataforma, além de facilitar a inserção dos alunos no mercado de trabalho por promover familiaridade com ferramentas reais, também contribui para uma melhor compreensão dos conteúdos abordados na disciplina.

Além de contribuir para o aprendizado prático dos conteúdos de Controle de Processos Industriais, a plataforma proposta também terá um papel relevante no suporte a futuros estudos, trabalhos e artigos acadêmicos desenvolvidos pelos alunos do campus. Ao oferecer um ambiente integrado para experimentação e documentação de resultados, a ferramenta poderá ser utilizada como base para investigações mais avançadas, permitindo que os estudantes explorem diferentes metodologias de controle e aplicações industriais, enriquecendo suas produções acadêmicas e ampliando o impacto educacional do projeto.

Portanto, o desenvolvimento deste trabalho se justifica pela necessidade de aprimorar a qualidade do ensino prático em automação, promovendo maior interação dos alunos com sistemas reais e possibilitando um aprendizado mais eficiente e aplicado. Dessa forma, a proposta visa preencher uma lacuna pedagógica essencial, promovendo um ensino mais eficaz, aplicado e alinhado às demandas da automação industrial contemporânea.

Para isso, são integrados diversos conhecimentos adquiridos ao longo da graduação, como os fundamentos de automação industrial, controle de processos, modelagem matemática de sistemas dinâmicos, instrumentação, além do uso de linguagens de programação específicas para CLPs, como Ladder e Texto Estruturado. Também são aplicados conceitos de sistemas supervisórios e análise de desempenho de controladores, o que exige uma abordagem multidisciplinar típica da formação em Engenharia de Controle e Automação.

### 1.3 Organização do Texto

Este trabalho está organizado em cinco capítulos, descritos a seguir:

- **Capítulo 1 – Introdução:** apresenta o contexto do estudo, a problemática enfrentada na disciplina de Controle de Processos Industriais, a justificativa para o desenvolvimento da plataforma, os objetivos geral e específicos, bem como a organização do texto.
- **Capítulo 2 – Revisão Bibliográfica:** reúne e discute os principais conceitos e fundamentos teóricos relacionados a sistemas de controle, controladores PID, compensadores de avanço e atraso de fase, discretização de controladores, CLPs e sistemas supervisórios com base na literatura especializada.
- **Capítulo 3 – Fundamentação Teórica:** aborda os conceitos fundamentais sobre sistemas de controle, controladores PID, compensadores de avanço de fase e atraso de fase, discretização de controladores, CLPs, sistemas supervisórios e ferramentas utilizadas no projeto.
- **Capítulo 4 – Metodologia:** descreve a abordagem adotada para o desenvolvimento da plataforma, incluindo a modelagem do sistema de tanques, métodos de discretização, estrutura do CLP, comunicação com o sistema supervisório e planejamento de testes.
- **Capítulo 5 – Desenvolvimento e Resultados:** apresenta a implementação da plataforma proposta, detalhando as etapas de desenvolvimento do sistema, a configuração do ambiente experimental, os testes realizados e a análise dos resultados obtidos.
- **Capítulo 6 – Conclusão:** sintetiza os principais resultados obtidos ao longo do trabalho, discutindo as contribuições da plataforma desenvolvida para o projeto e indicando possibilidades de trabalhos futuros na área de controle e automação.

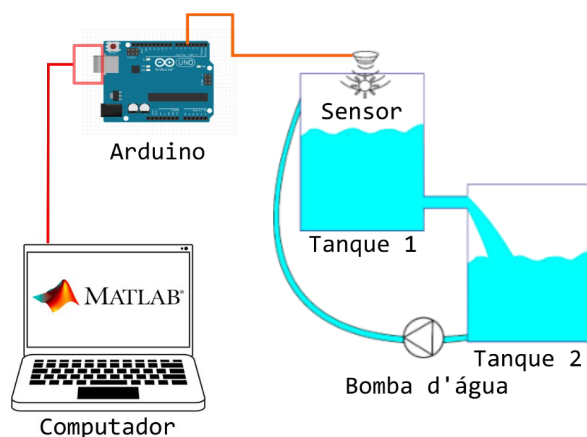
## 2 REVISÃO BIBLIOGRÁFICA

Com os avanços tecnológicos, especialmente a partir das revoluções da Indústria 4.0, surgiram novas demandas por integração entre sistemas físicos e digitais, destacando-se o uso de sensores inteligentes, redes industriais, Internet das Coisas (IoT, do inglês *Internet of Things*), computação em nuvem e análise de dados em tempo real (RIBEIRO; ABREU, 2020). Esse novo contexto exige profissionais com formação sólida e multidisciplinar, capazes de atuar de forma prática e teórica no desenvolvimento e implementação de soluções automatizadas.

Essa abordagem voltada ao ensino prático também é observada em pesquisas publicadas em periódicos e eventos da área. Por exemplo, Sauer *et al.* (2017) desenvolveram uma plataforma didática com implementação de controlador PID e interface gráfica, avaliando ganhos significativos na assimilação dos conceitos pelos estudantes. Já Amaral *et al.* (2011) apresentaram um ambiente para simulação e aplicação de controladores PID com anti *wind-up*, utilizando LabVIEW e uma planta de nível real, incentivando os alunos a programarem seus próprios algoritmos de controle.

Diversos trabalhos acadêmicos têm buscado criar bancadas didáticas e protótipos com foco no ensino prático de controle de processos. Um exemplo é o trabalho de Pimentel (2024), que propôs o desenvolvimento de um protótipo de controle de nível com dois tanques acoplados. O sistema foi construído com baixo custo utilizando Arduino, sensor ultrassônico e interface com o MATLAB para implementação de um controlador PID. O trabalho destaca-se pela preocupação com a acessibilidade do projeto, uma vez que kits comerciais podem ser inacessíveis para instituições públicas. Além disso, o sistema permite a análise prática do desempenho do controlador PID com base na resposta do nível de água em tempo real. A Figura 2 apresenta o diagrama do sistema idealizado elaborado pela autora.

Figura 2 – Diagrama do sistema idealizado.



Fonte: (PIMENTEL, 2024).

Outro projeto relevante foi desenvolvido por Hotz (2014), que elaborou uma bancada didática voltada ao ensino de controle de nível e temperatura, utilizando um CLP da fabricante

Schneider Electric, sensores analógicos, e uma Interface Homem-Máquina (IHM). A programação foi feita em linguagem Ladder com auxílio do *software* TwidoSuite, e a interface supervisória foi construída em Eclipse. A bancada permitiu aos alunos observarem respostas do sistema a diferentes entradas e realizar ajustes nos parâmetros dos controladores PI e *On/Off*, promovendo um aprendizado efetivo da dinâmica dos sistemas industriais simulados. A Figura 3 apresenta a bancada didática elaborada pela autora.

Figura 3 – Bancada didática para controle de nível e temperatura.



Fonte: (HOTZ, 2014).

Complementando essa abordagem, Ferreira (2016) trabalhou com o desenvolvimento de um sistema de controle discreto para supervisão de processos simulados. O projeto utilizou uma IHM *touchscreen* conectada a um CLP Siemens via rede Profibus DP, permitindo o controle de duas plantas virtuais. O autor implementou a lógica de controle em Ladder e desenvolveu interfaces gráficas na IHM para operação manual e automática, além de uma interface de ajuste dos parâmetros do controlador. Embora o foco tenha sido em simulação, o projeto se destaca pela integração de elementos industriais autênticos, como redes de comunicação e plataformas comerciais, o que proporciona ao aluno uma experiência mais próxima da realidade do ambiente industrial. A Figura 4 apresenta a tela da IHM desenvolvida pelo autor para a supervisão do processo e o ajuste dos parâmetros do controlador PID.

Figura 4 – Tela de supervisão do controlador PID.



Fonte: (FERREIRA, 2016).

Esses trabalhos demonstram que há um movimento consistente na área acadêmica para aproximar os estudantes da prática de controle de processos, seja por meio de bancadas físicas de baixo custo, seja com o uso de simulações realistas integradas a equipamentos industriais. Contudo, observa-se que a maioria dos projetos ainda se limita à aplicação do controlador PID e não explora amplamente outras estratégias como os compensadores de avanço e atraso de fase, nem oferece meios comparativos diretos entre essas técnicas.

Diante disso, o presente trabalho propõe uma plataforma didática que permite a implementação e comparação de controladores PID, de avanço de fase e de atraso de fase. Utilizando um CLP da Altus, o *software* MasterTool IEC XE e o supervisório BluePlant, a proposta busca oferecer uma solução completa, realista e acessível para o ensino de controle de processos, propondo uma abordagem distinta das observadas nos trabalhos anteriores e ampliando as possibilidades de aprendizado prático.

### 3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os principais fundamentos teóricos necessários para a compreensão do desenvolvimento da plataforma didática proposta. São apresentados os princípios de sistemas de controle, os controladores PID, de avanço e de atraso de fase, a linguagem de programação em texto estruturado, os sistemas de supervisão e IHM.

#### 3.1 Sistemas de controle

Os sistemas de controle classificam-se em malha aberta e malha fechada, conforme a existência de realimentação. A utilização da realimentação permite corrigir desvios e melhorar o desempenho do sistema frente a perturbações (MAYA; LEONARDI, 2014).

Em um sistema de malha aberta, o sinal de controle é gerado com base apenas na entrada de referência, sem considerar a saída medida do sistema. Não há realimentação, ou seja, o sistema não compara o valor de saída com o valor desejado. A ação de controle é aplicada supondo que o processo irá se comportar como esperado (DORF; BISHOP, 2020). Segundo Ogata (2010), a principal desvantagem do controle em malha aberta é a incapacidade de corrigir perturbações externas ou mudanças na dinâmica do processo.

Por outro lado, nos sistemas de malha fechada, também conhecidos como sistemas com realimentação, a saída do processo é continuamente monitorada e comparada com um valor de referência. A diferença entre esses dois valores, chamada de erro, é utilizada pelo controlador para ajustar o sinal de controle, com o objetivo de reduzir esse erro e manter a saída o mais próximo possível da referência desejada (DORF; BISHOP, 2020).

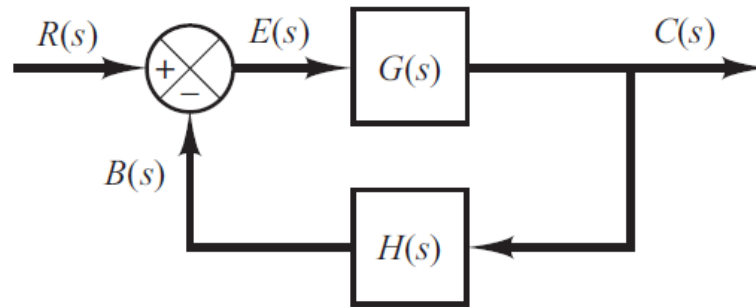
Em sistemas de controle em tempo discreto, o erro é definido como a diferença entre o sinal de referência e a saída do sistema no instante de amostragem  $k$ , sendo expresso por:

$$e(k) = r(k) - c(k), \quad (3.1)$$

em que  $r(k)$  é o sinal de referência e  $c(k)$  é a saída do sistema no instante  $k$ . Esse erro é utilizado nas leis de controle dos controladores digitais para o cálculo do sinal de controle  $u(k)$ .

Esse tipo de sistema é composto por diversos elementos, como pode ser observado na Figura 5. O sinal de referência é representado por  $R(s)$ , ou seja, o valor desejado que se pretende alcançar. Esse valor é comparado com o sinal de realimentação  $B(s)$ , proveniente da saída do processo, resultando em um sinal de erro  $E(s)$ , que indica a diferença entre o valor desejado e o valor real obtido. O erro é então aplicado à função de transferência do controlador e da planta  $G(s)$ , responsável por gerar a saída do sistema  $C(s)$ . Essa saída, por sua vez, passa por um bloco de realimentação  $H(s)$ , e retorna como sinal de realimentação  $B(s)$ , fechando o laço de controle (OGATA, 2010).

Figura 5 – Sistema de malha fechada.



Fonte: (OGATA, 2010).

A função de transferência em malha fechada, que define a relação entre a entrada  $R(s)$  e a saída  $C(s)$ , é expressa por (OGATA, 2010):

$$T(s) = \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (3.2)$$

Esse modelo matemático evidencia como a realimentação afeta o comportamento do sistema, permitindo, na maioria dos casos, respostas confiáveis mesmo diante de perturbações e variações na dinâmica do processo.

### 3.1.1 Sistemas de segunda ordem e parâmetros de desempenho

Grande parte dos sistemas de controle contínuos pode ser representada por modelos lineares invariantes no tempo (LIT). Em muitos casos, sistemas de ordem superior podem ser aproximados por um modelo de segunda ordem dominante, pois sua dinâmica é governada principalmente por um par de polos complexos conjugados. Essa aproximação permite relacionar a posição dos polos no plano- $s$  com o comportamento da resposta transitória, sendo amplamente utilizada na análise e no projeto de controladores (OGATA, 2010; AGUIRRE, 2020).

A forma padrão da função de transferência de um sistema linear invariante no tempo de segunda ordem é dada por (OGATA, 2010):

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \quad (3.3)$$

em que  $\omega_n$  é a frequência natural não amortecida do sistema e  $\zeta$  é o fator de amortecimento. Esses dois parâmetros estão diretamente relacionados à posição dos polos no plano- $s$ , os quais podem ser escritos como (OGATA, 2010):

$$s = -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2}. \quad (3.4)$$

A frequência natural  $\omega_n$  está associada à rapidez da resposta do sistema, enquanto o fator de amortecimento  $\zeta$  indica o grau de oscilação. Para  $0 < \zeta < 1$ , o sistema é classificado como

subamortecido e apresenta sobressinal e oscilações amortecidas na resposta ao degrau. Para  $\zeta = 1$ , o sistema é criticamente amortecido, e para  $\zeta > 1$ , é superamortecido.

A partir desses parâmetros, podem ser definidos indicadores clássicos de desempenho no domínio do tempo. O máximo sobressinal  $M_p$ , para sistemas subamortecidos, é dado por (OGATA, 2010):

$$M_p = e^{\left(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \cdot 100\%, \quad (3.5)$$

e o tempo de acomodação  $t_s$ , considerando o critério de 2%, pode ser aproximado por (OGATA, 2010):

$$t_s \approx \frac{4}{\zeta\omega_n}. \quad (3.6)$$

Esses parâmetros permitem avaliar quantitativamente o desempenho transitório de sistemas em malha fechada, sendo amplamente utilizados no projeto e na análise de controladores, especialmente em aplicações industriais de controle de processos.

## 3.2 Modelagem Matemática de Sistemas

A modelagem matemática é essencial para descrever o comportamento dinâmico do sistema e, posteriormente, projetar controladores eficazes. Ao modelar um sistema físico, é necessário compreender como suas variáveis de entrada influenciam as variáveis de saída. Essa modelagem pode ser realizada utilizando duas abordagens principais, que são o modelo de Caixa Branca e o modelo de Caixa Preta.

No modelo de Caixa Branca, o comportamento interno do sistema é conhecido e descrito por leis físicas ou equações matemáticas. Por exemplo, um tanque com entrada e saída de líquido pode ser modelado pela equação da conservação de massa. Esse tipo de modelo exige conhecimento profundo sobre os fenômenos envolvidos (AGUIRRE, 2007).

Já o modelo de Caixa Preta é utilizado quando não se conhece o funcionamento interno do sistema, e a modelagem é feita com base em observações experimentais. Nesse caso, um modelo matemático é ajustado com base em dados de entrada e saída, sem considerar diretamente a física do processo (AGUIRRE, 2007).

## 3.3 Controlador PID

O controlador PID é uma das estratégias de controle mais amplamente utilizadas na indústria devido à sua simplicidade, versatilidade e eficácia em uma ampla gama de sistemas dinâmicos. Sua função é ajustar a variável de controle de forma a reduzir o erro entre a saída do sistema e o valor de referência, por meio da combinação de três ações: proporcional, integral e derivativa (DORF; BISHOP, 2020).

A função de transferência do controlador PID pode ser expressa como (DORF; BISHOP, 2020):

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right), \quad (3.7)$$

onde:

- $K_p$  é o ganho proporcional,
- $T_i$  é o tempo integral (em segundos),
- $T_d$  é o tempo derivativo (em segundos),
- $s$  é a variável complexa de Laplace.

Essa estrutura permite o ajuste direto dos parâmetros de atuação do controlador, facilitando a sintonia conforme o comportamento desejado para o sistema. A ação proporcional reage ao erro presente, a ação integral busca eliminar o erro acumulado ao longo do tempo, e a ação derivativa antecipa a resposta com base na taxa de variação do erro.

Em muitos casos práticos, é comum a utilização do controlador Proporcional-Integral (PI), que elimina a componente derivativa. Essa simplificação é adequada para sistemas onde a ação derivativa pode amplificar o ruído da medição ou não é necessária (DORF; BISHOP, 2020).

A função de transferência do controlador PI é dada por (DORF; BISHOP, 2020):

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} \right). \quad (3.8)$$

O controlador PI oferece uma boa compensação entre desempenho e robustez, sendo amplamente aplicado em processos industriais que exigem eliminação do erro em regime permanente, sem a sensibilidade excessiva da ação derivativa.

### 3.4 Compensadores de Avanço de Fase e Atraso de Fase

Os compensadores de avanço de fase e atraso de fase são utilizados para alterar o comportamento do sistema, especialmente em termos de margem de fase, robustez e rejeição a perturbações. Esses compensadores são implementados por meio de redes de primeira ordem e sua função de transferência é projetada com base em critérios de desempenho em frequência ou lugar das raízes (OGATA, 2010).

Existem duas abordagens principais para o projeto desses compensadores: (i) o método do lugar das raízes e (ii) o método baseado na resposta em frequência. Ambos utilizam a mesma forma de função de transferência do compensador, porém a determinação de seus parâmetros é realizada de forma distinta em cada abordagem. Neste trabalho, optou-se pela utilização do

método do lugar das raízes, uma vez que ele permite analisar graficamente a influência do ganho e dos polos do compensador sobre a dinâmica do sistema em malha fechada.

No método do lugar das raízes, realiza-se a análise gráfica da trajetória dos polos do sistema de malha fechada no plano- $s$ , à medida que o ganho do sistema varia de zero ao infinito. A posição dos polos no plano complexo está diretamente relacionada ao comportamento dinâmico do sistema, como o tempo de acomodação ( $t_s$ ), o sobressinal ( $M_p$ ) e o fator de amortecimento ( $\zeta$ ) (OGATA, 2010).

Define-se inicialmente a posição desejada dos polos dominantes com base nos requisitos de desempenho. A posição  $s$  desses polos é calculada usando:

$$s = \sigma \pm j\omega_d, \quad \sigma = -\zeta\omega_n, \quad \omega_d = \omega_n\sqrt{1 - \zeta^2} \quad (3.9)$$

Com  $s$  conhecido, utiliza-se o critério do ângulo para verificar se esse ponto pertence ao lugar das raízes. Se não pertencer, adiciona-se um compensador com função de transferência:

$$G_c(s) = K \frac{s+z}{s+p} \quad (3.10)$$

onde  $K$  é o ganho do compensador,  $z$  representa a posição do zero e  $p$  a posição do polo do compensador no plano- $s$ , de forma que o zero e o polo sejam posicionados de modo a fazer com que o ponto  $s$  satisfaça simultaneamente os critérios de ângulo e de módulo do método do lugar das raízes:

$$\angle G(s)H(s) + \angle G_c(s) = \pm 180^\circ(2k+1), \quad k = 0, 1, \dots \quad (3.11)$$

Esse processo permite ajustar a trajetória do lugar das raízes para que os polos se posicionem nas localizações desejadas.

### 3.4.1 Compensador de Avanço de Fase

O compensador de avanço de fase é utilizado quando se deseja aumentar a margem de fase, elevar a frequência de cruzamento e, conseqüentemente, melhorar o tempo de subida e a estabilidade relativa do sistema. Isso o torna ideal para aplicações que exigem respostas mais rápidas e precisas, com menor atraso temporal e maior capacidade de rastreamento de sinais de referência.

A função de transferência de um compensador de avanço de fase pode ser representada de forma geral, conforme apresentado na Eq. (3.10). No projeto realizado pelo método da resposta em frequência, essa mesma função é frequentemente parametrizada da seguinte maneira:

$$G_c(s) = K \frac{Ts+1}{\alpha Ts+1}, \quad 0 < \alpha < 1 \quad (3.12)$$

onde:

- $K$  é o ganho do compensador,
- $T$  é a constante de tempo associada ao zero do compensador ( $z = -1/T$ ),
- $\alpha$  é o fator de avanço, tal que  $0 < \alpha < 1$ , e o polo é dado por  $p = -1/(\alpha T)$ .

Essa parametrização evidencia a relação entre as duas formas: o zero ( $z$ ) e o polo ( $p$ ) são posicionados de modo que o compensador introduza um avanço de fase dentro de uma determinada faixa de frequência específica. Isso proporciona maior velocidade de resposta e redução do tempo de subida sem comprometer a estabilidade.

### 3.4.2 Compensador de Atraso de Fase

O compensador de atraso de fase é indicado para situações em que o sistema apresenta pouca margem de ganho ou é suscetível a ruídos de alta frequência. Seu principal objetivo é aumentar a robustez, melhorar a rejeição a distúrbios de baixa frequência e permitir maior ganho em malha aberta, sem comprometer a estabilidade.

Assim como o compensador de avanço, sua função de transferência geral pode ser expressa conforme a Eq. (3.10). No método baseado na resposta em frequência, costuma-se utilizar a seguinte parametrização (OGATA, 2010):

$$G_c(s) = K \frac{Ts + 1}{\beta Ts + 1}, \quad \beta > 1 \quad (3.13)$$

onde:

- $K$  é o ganho do compensador,
- $T$  é a constante de tempo do zero ( $z = -1/T$ ),
- $\beta$  é o fator de atraso, tal que  $\beta > 1$ , e o polo é dado por  $p = -1/(\beta T)$ .

Esse compensador introduz um atraso de fase moderado, porém permite aumentar o ganho em baixas frequências, o que melhora o desempenho em regime permanente, como o erro de estado estacionário. O compensador de atraso de fase é frequentemente usado em sistemas industriais onde a estabilidade e a rejeição a perturbações são mais importantes que a rapidez da resposta, como em processos térmicos, hidráulicos ou de controle de velocidade de motores sob carga variável.

## 3.5 Sistemas no domínio do tempo contínuo e discreto

No controle de processos, os sistemas podem ser classificados no domínio do tempo contínuo ou discreto, de acordo com a forma como o tempo é tratado nas variáveis do sistema.

Nos sistemas contínuos, as variáveis de entrada e saída são definidas continuamente no tempo. Esse é o modelo típico de processos físicos reais, como variações de temperatura, nível de líquidos ou velocidade de motores. Os sinais são representados por funções contínuas do tempo, como  $x(t)$  ou  $y(t)$ , e suas dinâmicas são descritas por equações diferenciais.

Já nos sistemas discretos, as variáveis são definidas apenas em instantes específicos de tempo, geralmente obtidas por meio de amostragem periódica. Esses sistemas são comuns em ambientes digitais, nos quais sensores, controladores e atuadores interagem com sinais em intervalos de tempo definidos. Suas dinâmicas são representadas por equações de diferenças, que descrevem a relação entre estados e saídas em instantes discretos de tempo (DORF; BISHOP, 2020), e são expressas em função do tempo discreto  $k$ , como  $x[k]$  ou  $y[k]$ .

### 3.5.1 Discretização pelo Método de Tustin

O método de Tustin, também conhecido como transformação bilinear, é uma das técnicas mais utilizadas para converter sistemas do domínio contínuo para o domínio discreto. Essa transformação é especialmente relevante em projetos de controladores digitais, pois preserva importantes características da dinâmica do sistema original, como estabilidade e resposta transitória (AGUIRRE, 2020).

A substituição bilinear, que é apresentada abaixo, é baseada na aproximação do operador diferencial contínuo  $s$  por uma função racional em  $z$ , dada por (AGUIRRE, 2020):

$$s \approx \frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \quad (3.14)$$

ou, de forma algébrica equivalente:

$$s \approx \frac{2}{T} \cdot \frac{z - 1}{z + 1} \quad (3.15)$$

onde:

- $T$  é o período de amostragem (tempo entre cada amostra).

Essa substituição permite transformar uma função de transferência contínua  $G(s)$  em uma função equivalente no domínio discreto  $G(z)$ , viabilizando sua implementação em controladores digitais.

Após obter a função de transferência do controlador discretizada, tem-se:

$$G(z) = \frac{U(z)}{E(z)} \quad (3.16)$$

onde:

- $U(z)$  é a transformada Z do sinal de controle  $u(k)$ ;

- $E(z)$  é a transformada Z do erro  $e(k)$ .

A função  $G(z)$  terá a forma racional (AGUIRRE, 2020):

$$G(z) = \frac{b_0 + b_1z^{-1} + \dots + b_nz^{-n}}{1 + a_1z^{-1} + \dots + a_mz^{-m}} \quad (3.17)$$

A partir da função de transferência apresentada na Eq. (3.16), multiplicando ambos os lados por  $E(z)$ , tem-se:

$$U(z) = G(z) \cdot E(z) \quad (3.18)$$

$$U(z) (1 + a_1z^{-1} + \dots + a_mz^{-m}) = (b_0 + b_1z^{-1} + \dots + b_nz^{-n}) E(z) \quad (3.19)$$

Aplicando a transformada inversa de Z, obtém-se a equação no tempo discreto (equação de diferenças):

$$u(k) + a_1u(k-1) + a_2u(k-2) + \dots + a_mu(k-m) = b_0e(k) + b_1e(k-1) + b_2e(k-2) + \dots + b_ne(k-n) \quad (3.20)$$

Isolando  $u(k)$ , a equação pode ser reescrita na forma geral:

$$u(k) = -a_1u(k-1) - a_2u(k-2) - \dots - a_mu(k-m) + b_0e(k) + b_1e(k-1) + b_2e(k-2) + \dots + b_ne(k-n) \quad (3.21)$$

onde  $m$  e  $n$  são as ordens máximas dos termos da saída e da entrada, respectivamente. Os coeficientes  $a_i$  e  $b_j$  caracterizam a dinâmica do sistema discreto.

A equação Eq. (3.21) descreve a implementação prática do controlador no domínio discreto, permitindo o cálculo do sinal de controle  $u(k)$  a cada instante de amostragem, com base nos erros passados e nos sinais de controle anteriores.

### 3.6 Sistemas Supervisórios

Os sistemas supervisórios são plataformas de *software* utilizadas para monitorar, controlar e registrar dados de processos industriais em tempo real. Esses sistemas são, em geral, implementados por meio de arquiteturas do tipo SCADA, que integram funções de supervisão, controle e aquisição de dados. Eles processam e exibem informações de forma clara, auxiliando os operadores e engenheiros na tomada de decisão. Segundo Moraes e Castrucci (2010), a Engenharia de Automação Industrial depende fortemente da capacidade de monitorar e intervir nos processos, tornando os sistemas supervisórios ferramentas indispensáveis.

Além de permitir o acompanhamento de variáveis e o registro histórico, os sistemas SCADA possibilitam o gerenciamento de alarmes e o diagnóstico de falhas, desempenhando

papel essencial na integração entre níveis operacionais e corporativos (RIBEIRO; ABREU, 2020). As IHMs são componentes fundamentais desses sistemas, pois estabelecem o elo direto entre o operador e o processo. Conforme Ideali (2021), a evolução das IHMs e das redes industriais conectadas a sistemas supervisórios tem favorecido a criação de ambientes industriais mais inteligentes e interconectados.

A arquitetura típica de um sistema supervisório é composta por:

- Dispositivos de campo (sensores, atuadores, válvulas, motores, etc.);
- CLPs, responsáveis por controlar localmente os processos;
- *Software* supervisório (SCADA), executado em IHMs ou computadores industriais;
- Redes industriais, que garantem a troca de dados entre os níveis.

### 3.6.1 Comunicação e Protocolos de Rede

A comunicação entre o sistema supervisório e CLP é um dos elementos centrais em sistemas de automação industrial. Essa comunicação é realizada por meio de protocolos de redes industriais, responsáveis pela estruturação, endereçamento, integridade e sincronização dos dados trocados entre os dispositivos (IDEALI, 2021). Segundo Moraes e Castrucci (2010), a confiabilidade da comunicação é fator essencial para garantir a operação contínua e segura dos processos automatizados, pois qualquer falha na troca de dados pode comprometer o desempenho do sistema de controle.

Os protocolos industriais definem regras e padrões que permitem a interoperabilidade entre dispositivos de diferentes fabricantes, integrando sensores, atuadores, CLPs e *softwares* supervisórios em um mesmo ambiente de automação. Essa padronização está alinhada aos princípios da Indústria 4.0, que enfatiza a integração de sistemas ciberfísicos, conectividade e monitoramento em tempo real (RIBEIRO; ABREU, 2020; LUGLI; SANTOS, 2023). De acordo com Ideali (2021), a convergência entre tecnologias de automação e redes de comunicação baseadas em Ethernet representa um dos pilares da Indústria 4.0, possibilitando o uso de protocolos abertos e interoperáveis para comunicação entre dispositivos industriais e plataformas digitais. Nesse contexto, destacam-se alguns dos principais protocolos de comunicação industrial utilizados para integrar dispositivos de campo, CLPs, sistemas supervisórios e plataformas digitais, conforme descrito a seguir:

- Modbus RTU: protocolo serial baseado em RS-485, amplamente empregado em pequenas redes. Possui estrutura simples e baixo custo de implementação, sendo ideal para comunicações ponto a ponto ou em topologia de barramento;
- Modbus TCP/IP: evolução do Modbus RTU, utiliza a rede Ethernet como meio físico, permitindo maiores taxas de transmissão e fácil integração com redes corporativas. É um

dos protocolos mais compatíveis com sistemas supervisórios como o BluePlant (ALTUS S.A., 2018);

- OPC (*OLE for Process Control*): protocolo padronizado baseado na arquitetura Windows, responsável por facilitar a interoperabilidade entre softwares e equipamentos de diferentes fabricantes;
- Profibus: protocolo desenvolvido na Alemanha, amplamente utilizado em sistemas Siemens. Baseia-se em redes seriais e oferece comunicação determinística e confiável em ambientes industriais hostis;
- Profinet: evolução do Profibus, utiliza redes Ethernet industriais, proporcionando altas velocidades de comunicação e integração com tecnologias de TI, o que o torna adequado a aplicações em tempo real e sistemas distribuídos;
- Ethernet/IP: protocolo desenvolvido pela Rockwell Automation que utiliza o padrão Ethernet industrial, permitindo comunicação entre dispositivos Allen-Bradley e integração com sistemas SCADA e IHMs em diferentes camadas de rede.

Esses protocolos garantem que o sistema supervisório acesse variáveis de processo em tempo real, viabilizando controle e supervisão contínuos. Conforme Maya e Leonardi (2014), a comunicação eficiente entre os níveis de campo e supervisão permite estratégias de controle mais avançadas. A tendência atual é a adoção de protocolos baseados em Ethernet, como o Modbus TCP/IP e o Profinet, que facilitam a integração entre os níveis operacional e corporativo, promovendo conectividade e interoperabilidade (LUGLI; SANTOS, 2023).

### 3.6.2 Tags

Nos sistemas supervisórios, a interface entre o processo físico e o ambiente de supervisão é realizada por meio das *tags*, variáveis simbólicas que representam dados do processo, como leituras de sensores, estados de atuadores ou valores internos de cálculo. Elas são fundamentais para a comunicação entre o CLP e o supervisório, permitindo a troca estruturada de informações.

As *tags* podem ser classificadas em:

- *Tags* de leitura (*Input Tags*): trazem informações do processo para o supervisório;
- *Tags* de escrita (*Output Tags*): enviam comandos do supervisório para o processo;
- *Tags* internas: usadas apenas no supervisório, para cálculos, alarmes ou lógicas auxiliares.

Segundo Ideali (2021), a correta definição e estruturação das *tags* é essencial para garantir confiabilidade e desempenho, pois elas funcionam como elo entre o nível de controle e a camada

de supervisão, possibilitando monitoramento, comando e registro eficientes das variáveis de processo.

### **3.7 Linguagem de Programação em Texto Estruturado**

A linguagem de Texto Estruturado do inglês, Structured Text (ST) é uma das linguagens de programação padronizadas pela norma IEC 61131-3, que define cinco linguagens utilizadas em CLPs, sendo o Texto Estruturado uma das mais adequadas para a implementação de algoritmos matemáticos e funções complexas (PLCopen, 2004).

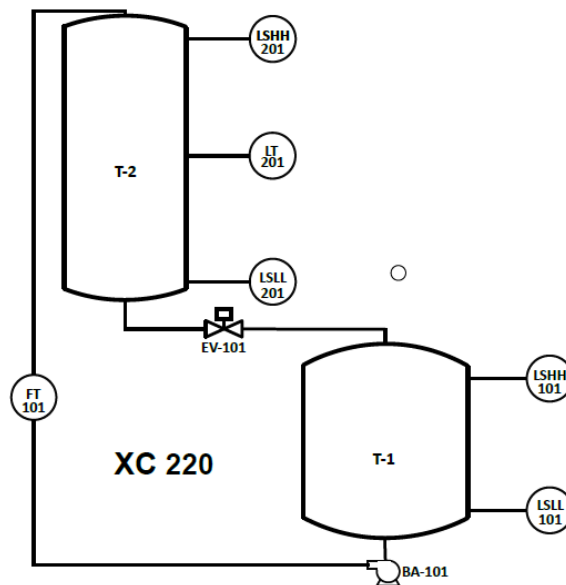
Ela se assemelha a linguagens de alto nível como Pascal ou C, e permite o desenvolvimento de algoritmos complexos, estruturas condicionais, laços e o uso de funções matemáticas. Por isso, é ideal para a implementação de algoritmos de controle, como os de avanço, atraso de fase e PID.

Enquanto linguagens como Ladder são visualmente intuitivas para lógicas discretas, a ST se destaca pela clareza e escalabilidade em algoritmos matemáticos, possibilitando modularidade e reutilização de código em aplicações industriais reais. No ambiente de programação MasterTool IEC XE, utilizado com os CLPs da Altus, a linguagem ST é amplamente empregada para a implementação de controladores contínuos e estruturas orientadas a objetos (ALTUS, 2021).

## 4 METODOLOGIA

Será utilizada uma bancada de ensaios para elaboração da plataforma proposta. Essa bancada consiste em um sistema de dois tanques acoplados, utilizado para o estudo de controle de processos contínuos. A Figura 6 apresenta o diagrama P&ID do sistema, no qual é representada, de forma esquemática, a configuração da planta didática. Nessa figura, são apresentados dois tanques, uma bomba de recirculação (BA-101), uma válvula solenoide de retorno (EV-101), sensores do tipo boia para detecção de nível alto e baixo (LSHH e LSLI), um transmissor de pressão piezoresistivo (LT-201) e um medidor de vazão tipo turbina (FT-101). A bomba será responsável por elevar a água do tanque inferior para o superior, enquanto a válvula controlará o retorno da água.

Figura 6 – Diagrama esquemático da bancada de ensaios.



Fonte: (Exsto Tecnologia Ltda, 2021).

Como mostrado no diagrama esquemático, a plataforma contém um transmissor de pressão piezoresistivo, sendo este do modelo RTP-420, da fabricante Rücken. Esse sensor converte a pressão exercida pelo fluido em um sinal elétrico proporcional a uma tensão entre 0 e 10 V, compatível com entradas analógicas padrão de CLPs. O modelo adotado opera em uma faixa de medição de 0 a 40 bar, com exatidão de 0,25% do fundo de escala (classe A3) e alimentação de 8 a 32 Vcc (Rücken Automação, 2020). Na Figura 7, observa-se o transmissor de pressão piezoresistivo.

Figura 7 – Transmissor de pressão piezoresistivo.



Fonte: Próprio autor.

O sistema de circulação de fluido da bancada é composto por uma bomba de recirculação e uma válvula solenoide. A bomba utilizada é uma minibomba centrífuga de corrente contínua, do tipo *brushless*, com alimentação de 24 Vcc. Sua função é promover o fluxo do líquido entre os tanques e os demais componentes do sistema, garantindo uma circulação contínua e estável. A Figura 8 apresenta a bomba instalada na bancada.

Figura 8 – Bomba de recirculação.



Fonte: Próprio autor.

A válvula solenoide empregada é do modelo BGH02413C, da fabricante WERK-SCHOTT

Pneumática, projetada para operar com tensão de 24 Vcc e potência de 9 W. Essa válvula atua no controle do fluxo do fluido, permitindo ou interrompendo sua passagem de forma automatizada, conforme comandos enviados pelo CLP. A Figura 9 exibe a válvula solenoide instalada na bancada.

Figura 9 – Válvula solenoide.



Fonte: Próprio autor.

Para a modelagem do sistema, serão coletados dados do transmissor de pressão piezo-resistivo, responsável por medir a pressão exercida pela coluna de líquido no tanque. Como os sinais são lidos em formato digital pelo CLP, será necessário aplicar uma calibração que relacione essas leituras com o nível real do fluido.

Esse processo será realizado utilizando o *software* Scilab, por meio da função `polyfit()`, que ajusta uma equação polinomial aos dados experimentais. Para este caso, será aplicada uma regressão linear de primeiro grau, resultando em uma equação do tipo  $h = a \cdot x + b$ , onde  $h$  é a altura do líquido e  $x$  é o valor digital lido pelo CLP. Essa equação permitirá estimar, em tempo real, a altura do fluido com base nos dados do sensor.

Após a obtenção da função de calibração dos dados do sensor, será dado início à elaboração do código na linguagem de Texto Estruturado no *software* MasterTool. Inicialmente, será implementada a função para o cálculo da altura real do fluido a partir das leituras recebidas pelo CLP. A partir da informação dos valores de início e fim do degrau desejado, o sistema executará a transição entre esses valores por meio da aplicação de um sinal PWM (*Pulse Width Modulation*) à bomba, registrando a resposta do nível do fluido ao longo do tempo.

Com os valores obtidos, é possível traçar a curva de resposta do sistema em malha aberta, a partir da qual será possível identificar parâmetros dinâmicos para um sistema de primeira ordem,

como o ganho estático  $K$  e a constante de tempo  $T$ . Com esses dados, será feita a modelagem da função de transferência do sistema em malha aberta, que se caracteriza como um modelo de Caixa Preta, já que o modelo será obtido a partir da resposta temporal do sistema a uma entrada em degrau unitário.

A partir da função de transferência de malha aberta, podem ser calculados os controladores de avanço de fase, atraso de fase e PID no domínio do tempo contínuo. Para a implementação desses controladores em Texto Estruturado, será gerada no código uma função de discretização, que converterá os controladores do domínio do tempo contínuo para o discreto por meio do método de Tustin e geração da equação de diferença.

Para isso, o usuário deverá informar os valores previamente calculados dos controladores. No caso dos controladores de avanço de fase e atraso de fase, deverão ser fornecidos os valores do ganho  $K$ , o zero e o polo. Para o controlador PID, deverão ser informados o ganho proporcional  $K_p$ , o tempo integral  $T_i$  e o tempo derivativo  $T_d$ . Após a discretização dos controladores, o código irá implementá-los no sistema, gerando a resposta ao degrau em malha fechada conforme o ponto de operação que será informado pelo usuário.

Para a supervisão do sistema, será desenvolvida uma tela de interface no *software* BluePlant, que será integrada ao CLP da Altus. Nessa tela, o usuário encontrará, inicialmente, as opções para executar as etapas dos experimentos com o objetivo de obter os dados da resposta do sistema em malha aberta, bem como a implementação dos controladores e a resposta em malha fechada.

A comunicação entre o BluePlant e o CLP será realizada via protocolo Modbus TCP, que permite a troca confiável de dados entre dispositivos. No BluePlant, serão criadas telas gráficas representando a planta, com indicadores, botões e elementos visuais vinculados às variáveis do processo. Para essa integração, serão utilizadas *s* de comunicação, que fazem a associação entre os elementos gráficos do supervisório e os endereços de memória do CLP.

Cada variável utilizada no programa do CLP (como o nível do tanque, o estado da bomba) será associada a uma *Tag* no supervisório, garantindo leitura e escrita em tempo real. Essa estrutura permitirá ao usuário interagir com o processo, definir parâmetros e acompanhar a resposta do sistema de forma intuitiva e eficiente.

Por fim, os resultados serão analisados com base em indicadores clássicos de desempenho de sistemas de controle, como tempo de subida, tempo de acomodação, sobressinal, erro em regime permanente, estabilidade e robustez. Além disso, os resultados obtidos na plataforma serão comparados com simulações dos mesmos controladores realizadas no Scilab, utilizando gráficos de resposta ao degrau e os mesmos indicadores de desempenho, permitindo uma análise comparativa entre o sistema real e o modelo simulado.

## 5 DESENVOLVIMENTO E RESULTADOS

Nesta etapa, são apresentadas as fases de construção e ensaios na bancada, destacando o processo de discretização dos controladores, a calibração do sensor de nível, a elaboração do código em Texto Estruturado, a configuração das saídas do CLP, desenvolvimento das telas do sistema supervisor e a comunicação entre os dispositivos.

### 5.1 Elaboração das Equações de Diferenças

Para a implementação dos controladores na prática, foram desenvolvidas as funções de discretização utilizando o método de Tustin, com o objetivo de realizar automaticamente a discretização dos controladores a partir dos dados no domínio contínuo informados pelo usuário. A partir dessa discretização, são obtidos os coeficientes necessários para a formação das equações de diferenças, que são então utilizados pelo programa em Texto Estruturado para o cálculo do sinal de controle  $u(k)$ . Esse valor é enviado à saída do CLP, que aciona a bomba de forma proporcional à ação de controle calculada. A seguir, são apresentadas as equações de diferenças utilizadas para o controlador PID e para os compensadores de avanço e atraso de fase.

#### 5.1.1 Controlador PID Discreto no Tempo

Aplicando o método de discretização de Tustin ao controlador PID, obtém-se a seguinte equação de diferenças para o cálculo do sinal de controle  $u(k)$ :

$$u(k) = u(k-1) + K_p [e(k) - e(k-1)] + K_i T_s e(k) + \frac{K_d}{T_s} [e(k) - 2e(k-1) + e(k-2)], \quad (5.1)$$

onde  $K_p$ ,  $K_i$ ,  $K_d$  são os ganhos proporcional, integral e derivativo, respectivamente,  $T_s$  é o período de amostragem, e  $e(k)$  representa o erro de rastreamento no instante  $k$ .

#### 5.1.2 Compensadores de Avanço de Fase e Atraso de Fase Discretos no Tempo

A função de transferência contínua dos compensadores de avanço de fase e atraso de fase, apresentada na forma racional na Eq. (3.17), é discretizada neste estudo pelo método de Tustin, de modo a possibilitar sua implementação digital no CLP.

A função de transferência dos compensadores de avanço de fase e atraso de fase no domínio do tempo discreto pode ser representada pelas equações de diferenças obtidas pela aplicação da transformada inversa de Z, expressas genericamente como:

$$u(k) = a_1 u(k-1) + a_2 u(k-2) + b_0 e(k) + b_1 e(k-1) + b_2 e(k-2). \quad (5.2)$$

Os coeficientes  $a_1$ ,  $a_2$ ,  $b_0$ ,  $b_1$  e  $b_2$  são obtidos a partir dos parâmetros do compensador contínuo e do período de amostragem  $T_s$ , segundo as expressões:

$$a_1 = \frac{8}{2T_s p + 4}, \quad (5.3)$$

$$a_2 = \frac{2T_s p - 4}{2T_s p + 4}, \quad (5.4)$$

$$b_0 = \frac{KT_s(T_s z + 2)}{2T_s p + 4}, \quad (5.5)$$

$$b_1 = \frac{2KT_s^2 z}{2T_s p + 4}, \quad (5.6)$$

$$b_2 = \frac{KT_s(T_s z - 2)}{2T_s p + 4}, \quad (5.7)$$

em que  $K$  é o ganho do compensador,  $z$  a posição do zero,  $p$  a posição do polo e  $T_s$  o período de amostragem.

## 5.2 Calibração do Sensor de Nível

Para que o CLP pudesse converter corretamente o valor do sinal analógico na medição de nível de líquido no tanque, foi necessário realizar a calibração do sensor do tipo transmissor de pressão piezoresistivo instalado na bancada. O sensor fornece uma tensão proporcional à altura da coluna de água no tanque, sendo essa tensão lida pela entrada analógica do CLP, configurada para leitura de tensão na escala de 0 a 10 V. O sinal recebido é convertido internamente pelo CLP em um valor numérico inteiro escalonado, que varia de 0 a 30000. Dessa forma, cada valor numérico corresponde a um determinado nível de líquido em milímetros, possibilitando a obtenção de uma relação direta entre a leitura digital e a altura real da água no tanque.

O procedimento de calibração consistiu em registrar os valores de leitura da entrada analógica do CLP para diferentes alturas do nível de água, na faixa de 20 a 460 mm, em incrementos de 20 mm, totalizando 23 medições. Os dados obtidos estão apresentados na Tabela 1, na qual se observa a correspondência entre o valor de leitura da entrada do CLP e o nível medido em milímetros.

Tabela 1 – Valores de calibração entre a leitura do CLP e o nível em milímetros.

Leitura do CLP	Nível (mm)
700	20
1250	40
2250	60
3200	80
4200	100
5350	120
6550	140
7600	160
8850	180
9950	200
11100	220
12300	240
13350	260
14500	280
15650	300
16900	320
17950	340
19250	360
20500	380
21550	400
22700	420
23950	440
25150	460

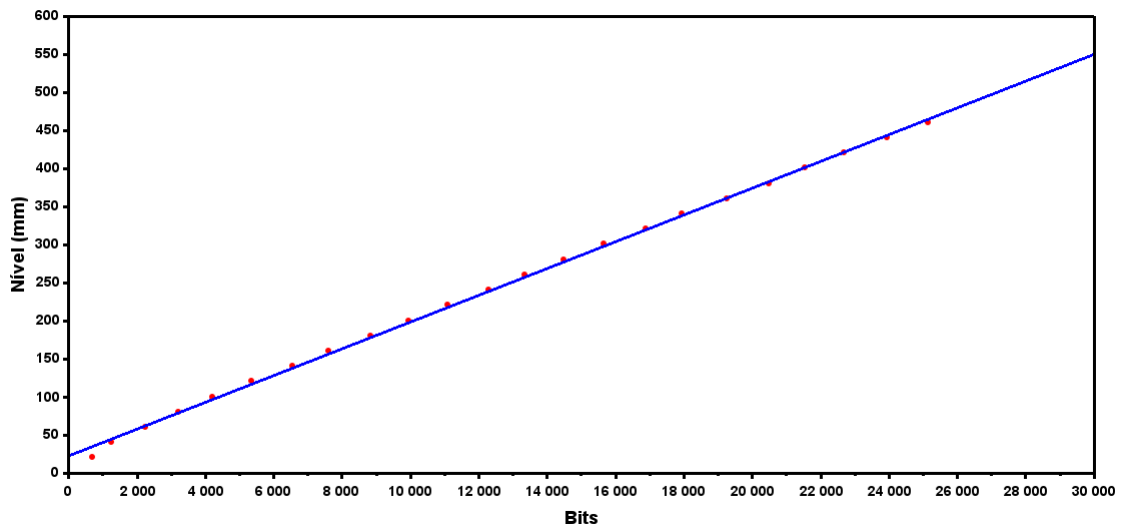
A partir dos dados coletados, aplicou-se uma regressão linear para determinar a equação que melhor representa a relação entre a leitura dos dados do sensor e o nível de água no tanque. No ajuste, foi obtido por meio de regressão linear, a variável  $x$  representa a leitura numérica correspondente à tensão da entrada analógica do CLP e a variável  $y$  o nível medido em milímetros. O resultado do ajuste é expresso pela equação:

$$y = 0,0176x + 22,3560 \quad (5.8)$$

Essa função descreve a relação afim entre o sinal de leitura do sensor e a altura do líquido no tanque, permitindo converter automaticamente os valores obtidos pelo CLP em milímetros. O coeficiente de determinação obtido foi  $R^2 = 0,9991$ , indicando um excelente ajuste do modelo aos dados experimentais. Assim, a calibração garante que as medições de nível realizadas durante os experimentos de controle sejam coerentes e representem com precisão o comportamento do sistema.

A Figura 10 apresenta o gráfico obtido a partir dos dados experimentais e da curva de ajuste. Os pontos em vermelho representam as medições realizadas, enquanto a linha contínua corresponde à função ajustada.

Figura 10 – Curva de ajuste.



Fonte: Próprio autor.

Com base nos dados apresentados no gráfico, observa-se que os valores medidos ficaram próximos aos dados obtidos por meio da Eq. (5.8), indicando que a equação de calibração obtida representa de forma adequada o comportamento do sensor para medição do nível do reservatório. Esse resultado comprova que o ajuste é satisfatório, uma vez que o coeficiente de determinação é muito próximo de 1.

### 5.3 Implementação do Código e Configuração da Saída de Controle

Após a geração das equações de diferenças e a calibração do sensor de nível, foi desenvolvido o algoritmo de controle responsável por processar as leituras de nível e gerar o sinal de saída para o acionamento da bomba.

Para realizar o acionamento da bomba, foi utilizada a saída digital Q14 do CLP Altus XP350, configurada em modo PWM (do inglês, *Pulse Width Modulation*). Essa escolha deve-se ao fato de o modelo do CLP utilizado não possuir saídas analógicas controláveis, o que inviabiliza o envio direto de um sinal de 0 a 24 V proporcional à ação de controle.

Nesse tipo de modulação, o tempo em que o sinal permanece em nível lógico alto, conhecido como ciclo ativo (do inglês, *duty cycle*), é ajustado de acordo com o sinal de controle calculado pelo algoritmo. Assim, quanto maior o valor de  $u(k)$ , maior será o tempo em que a saída permanece acionada durante cada ciclo, o que corresponde a uma maior potência média aplicada à bomba. Dessa forma, o PWM atua como uma forma digital de controle proporcional, permitindo regular a velocidade da bomba de maneira contínua mesmo utilizando apenas saídas digitais.

A rotina responsável pela configuração e controle da saída PWM no CLP converte o valor

calculado pelo controlador em um percentual, variando de 0 a 100%, que representa o ciclo ativo do sinal PWM. O valor da variável `Pot_bomba` é dividido por 30000, que corresponde ao valor máximo da leitura digital do CLP, e o resultado é multiplicado por 100 para obter a porcentagem equivalente. Em seguida, essa porcentagem é convertida para o tipo de dado `USINT` (*do inglês, Unsigned Short Integer*), por meio da função `REAL_TO_USINT`, tornando o valor compatível com a variável de saída utilizada pelo bloco de geração do sinal PWM.

O trecho de código que implementa essa rotina é apresentado a seguir.

Listing 5.1 – Trecho do código responsável pela configuração da saída PWM.

```
1 Cal_PWM := (VG.Pot_bomba / 30000) * 100;
2 Duty := REAL_TO_USINT(Cal_PWM);
3
4 PWM_Q14 (
5     ENABLE      := TRUE,
6     FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
7     FREQUENCY   := 1000,
8     DUTY_CYCLE  := Duty
9 );
```

O bloco `PWM_Q14` é então habilitado e configurado para operar na saída rápida Q14, com frequência de 1 kHz. O parâmetro `DUTY_CYCLE` recebe o valor calculado da variável `Duty`, determinando o tempo em que o sinal permanece ativo durante cada ciclo de modulação. Dessa forma, o CLP gera o sinal PWM proporcional à ação de controle, que é utilizado para acionar a bomba de forma contínua.

## 5.4 Desenvolvimento das Telas do Sistema Supervisório

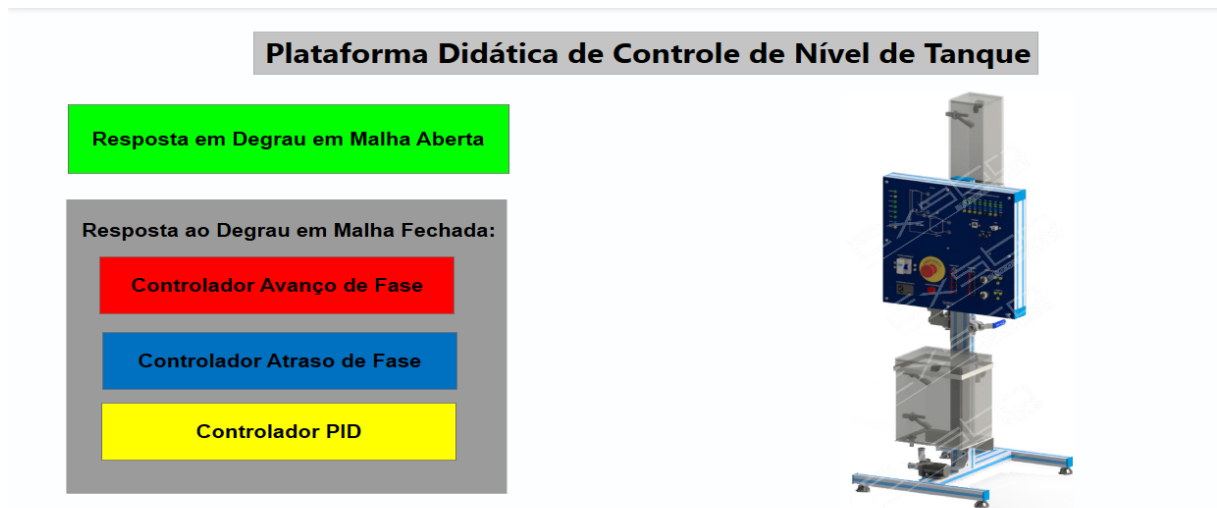
O sistema supervisório foi desenvolvido no software BluePlant, com o objetivo de permitir a interação do usuário com a bancada didática, a configuração dos parâmetros de controle e o acompanhamento visual dos testes realizados. O supervisório foi estruturado de forma intuitiva, possibilitando que qualquer usuário consiga operar o sistema sem conhecimentos avançados em automação. As telas foram organizadas em diferentes níveis, seguindo a lógica das funcionalidades implementadas no CLP.

### 5.4.1 Tela Inicial do Sistema

A primeira tela criada, apresentada na Figura 11, corresponde ao menu inicial do supervisório. Nela, o usuário encontra os principais botões de navegação, que direcionam para os diferentes tipos de testes disponíveis. Os botões foram organizados de maneira destacada, utilizando cores diferentes para facilitar a identificação visual das funções.

Nesta tela, o usuário pode escolher entre realizar a resposta ao degrau em malha aberta ou realizar testes de resposta ao degrau em malha fechada, onde é possível selecionar o controlador desejado: avanço de fase, atraso de fase ou PID. À direita da tela, foi adicionada uma imagem representativa da bancada, permitindo a rápida identificação do processo a ser controlado.

Figura 11 – Tela Inicial do Sistema.



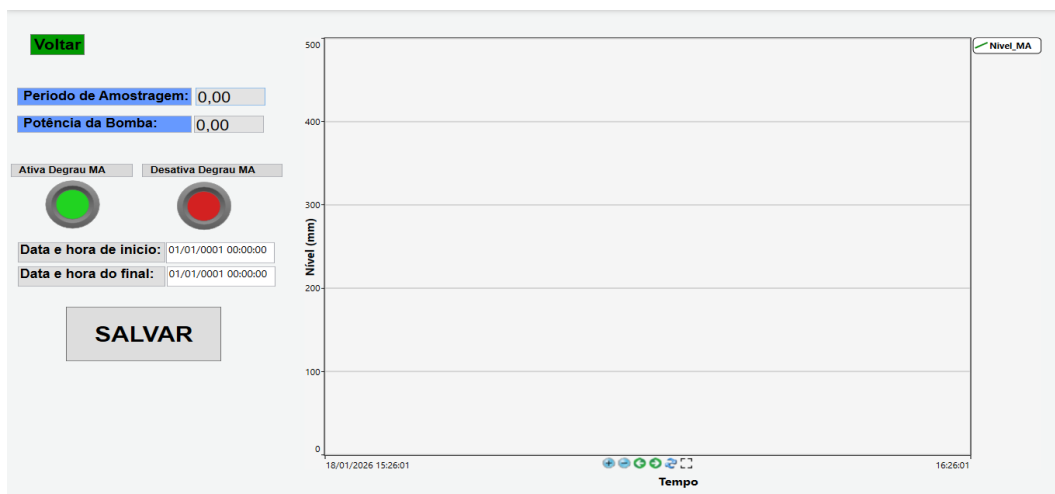
Fonte: Próprio autor.

#### 5.4.2 Tela de Resposta ao Degrau em Malha Aberta

A segunda tela apresentada na Figura 12 exibe a interface usada para realizar os testes de resposta ao degrau em malha aberta. Nela, o usuário pode informar o período de amostragem e potência aplicada à bomba. Esses parâmetros permitem configurar o ensaio, que será executado diretamente pelo CLP.

Foram incluídos botões para ativar e desativar a atuação do controlador na plataforma. A tela possui um gráfico que exibe em tempo real a resposta do sistema, permitindo ao usuário visualizar o comportamento do nível do tanque à medida que a bomba é acionada. Durante a realização do ensaio prático, existe a opção de registrar a data e hora de início e término das amostras, sendo possível salvar os dados adquiridos no período informado.

Figura 12 – Tela de Resposta ao Degrau em Malha Aberta.



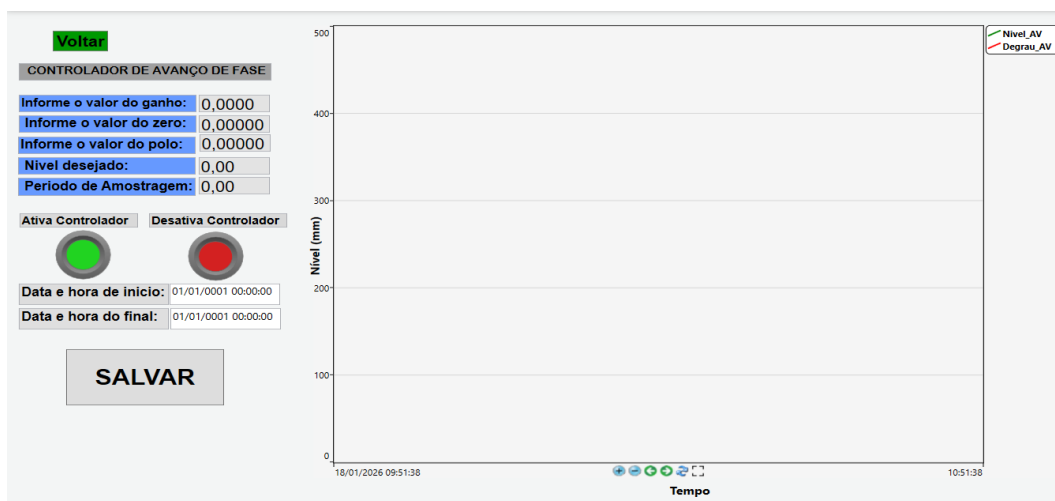
Fonte: Próprio autor.

### 5.4.3 Tela do Controlador de Avanço de Fase

A tela destinada ao controlador de avanço de fase apresentada na Figura 13 permite que o usuário insira os parâmetros necessários para a discretização e implementação do compensador pelo CLP. Entre os valores configuráveis estão o ganho, a posição do zero, a posição do polo, o nível desejado e o período de amostragem.

Essa tela também dispõe dos botões de ativação e desativação do controlador, e de um gráfico que exibe simultaneamente o nível real do tanque e o nível de referência enviado pelo degrau. Com isso, o usuário consegue visualizar a resposta do sistema em malha fechada. Há também a opção de salvar os dados do ensaio em uma planilha do Excel para facilitar a análise posterior.

Figura 13 – Tela de Resposta ao Controlador de Avanço de Fase.



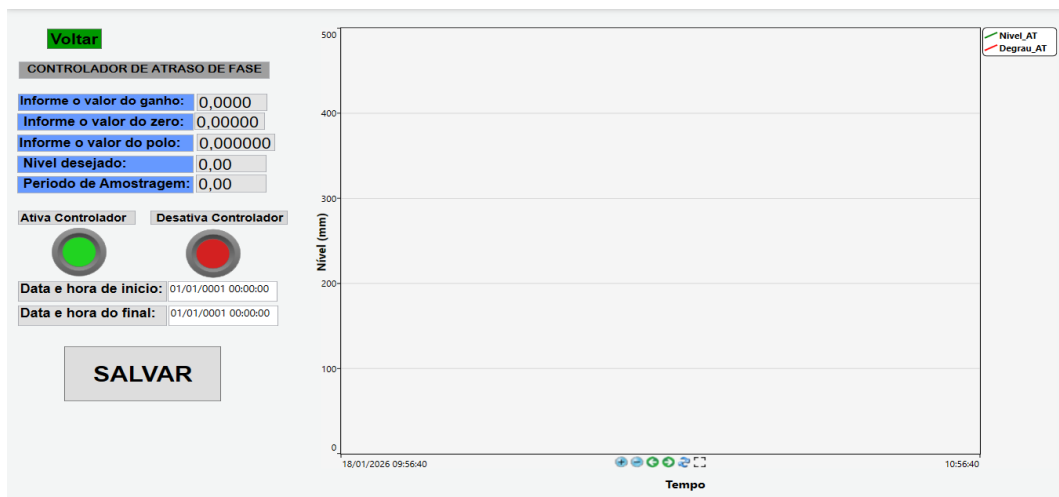
Fonte: Próprio autor.

#### 5.4.4 Tela do Controlador de Atraso de Fase

A interface do controlador de atraso de fase segue a mesma estrutura da tela anterior, apresentando campos para inserção dos parâmetros necessários ao cálculo dos coeficientes discretos. O usuário pode ajustar o ganho, zero, polo, nível desejado e o período de amostragem.

Da mesma forma, o gráfico à direita permite acompanhar o comportamento da malha fechada em tempo real. Os botões de ativação e desativação possibilitam iniciar ou interromper o controlador durante o teste, garantindo maior flexibilidade na operação. Além da opção salvar os dados do ensaio em uma planilha do Excel. A Figura 14 ilustra a interface de ensaio do controlador de Atraso de Fase.

Figura 14 – Tela de Resposta ao Controlador de Atraso de Fase.



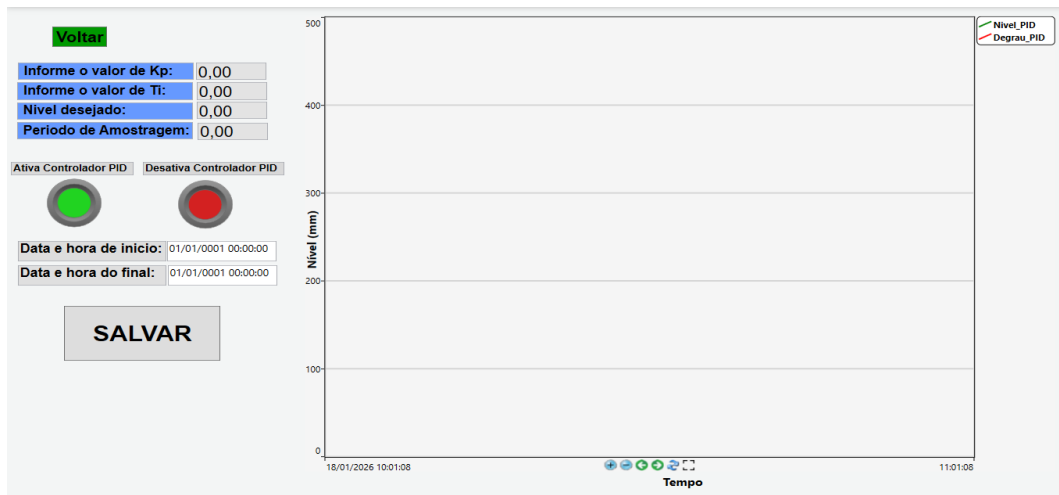
Fonte: Próprio autor.

#### 5.4.5 Tela do Controlador PID

A tela destinada ao controlador PID, apresentada na Figura 14, ilustra uma estrutura semelhante às demais telas de controle em malha fechada. Nela, o usuário informa os valores de  $K_p$  e  $T_i$ , além do nível desejado e do período de amostragem. Os botões possuem a mesma função, sendo também disponibilizada a opção de salvar os dados adquiridos.

O gráfico dedicado à resposta do nível do tanque mostra, em tempo real, o resultado da ação de controle, permitindo analisar a estabilidade, tempo de acomodação e o erro em regime permanente. Essa padronização das telas facilita o uso e diminui a possibilidade de erros por parte do operador.

Figura 15 – Tela do Controlador PID.



Fonte: Próprio autor.

## 5.5 Comunicação do Supervisório com o CLP

A comunicação entre o supervisório desenvolvido no BluePlant e o CLP Altus XP350 foi realizada por meio do protocolo Modbus TCP, utilizando a rede Ethernet para o envio e o recebimento de dados em tempo real. O CLP foi configurado como escravo Modbus, disponibilizando registradores destinados às variáveis usadas no controle, como nível do tanque, potência da bomba, parâmetros dos controladores e comandos de ativação dos testes.

No BluePlant, cada variável utilizada nas telas foi associada a uma *Tag*, mapeada para o registrador correspondente no CLP. Dessa forma, o supervisório consegue ler continuamente as informações do processo e enviar comandos sempre que o usuário altera algum parâmetro ou aciona um botão de teste. Esse mecanismo garante que o CLP e o supervisório permaneçam sincronizados durante toda a operação. A Tabela 2 apresenta a relação completa das *tags* utilizadas no supervisório, bem como seus respectivos endereços Modbus, tipos de dados e permissões de acesso.

Tabela 2 – Lista de *tags* do supervisor e seus respectivos endereços Modbus

Nome da TAG	Nó	Endereço	Tipo de Dado	Acesso	Modificadores
ATIVA_MA	modbus	1	Native	ReadWrite	–
DESATIVA_MA	modbus	2	Native	ReadWrite	–
ATIVA_AV	modbus	3	Native	ReadWrite	–
DESATIVA_AV	modbus	4	Native	ReadWrite	–
ATIVA_AT	modbus	5	Native	ReadWrite	–
DESATIVA_AT	modbus	6	Native	ReadWrite	–
ATIVA_PID	modbus	7	Native	ReadWrite	–
DESATIVA_PID	modbus	8	Native	ReadWrite	–
Nivel_MA	modbus	400001	Single	Read	WordSwap=True
Tempo_MA	modbus	400003	Single	Read	WordSwap=True
Degrau_MA	modbus	400005	Single	Write	WordSwap=True
Pot_bomba	modbus	400007	Native	Write	–
Nivel_AV	modbus	400008	Single	ReadWrite	WordSwap=True
Tempo_AV	modbus	400010	Single	Read	WordSwap=True
Degrau_AV	modbus	400012	Single	ReadWrite	WordSwap=True
POLO_AV	modbus	400014	Single	ReadWrite	WordSwap=True
ZERO_AV	modbus	400016	Single	ReadWrite	WordSwap=True
K_AV	modbus	400018	Single	ReadWrite	WordSwap=True
TS_AV	modbus	400020	Native	ReadWrite	–
Nivel_PID	modbus	400021	Single	ReadWrite	WordSwap=True
Tempo_PID	modbus	400023	Single	Read	WordSwap=True
Degrau_PID	modbus	400025	Single	ReadWrite	WordSwap=True
TS	modbus	400027	Native	ReadWrite	–
KP	modbus	400028	Single	Write	WordSwap=True
Ti	modbus	400030	Single	Write	WordSwap=True
BOMBA_AT	modbus	400032	Native	Read	–
BOMBA_AV	modbus	400034	Native	Read	–
BOMBA_PID	modbus	400036	Native	Read	–
TS_AT	modbus	400038	Native	ReadWrite	–
ZERO_AT	modbus	400040	Single	ReadWrite	WordSwap=True
POLO_AT	modbus	400042	Single	ReadWrite	WordSwap=True
K_AT	modbus	400044	Single	ReadWrite	WordSwap=True
Nivel_AT	modbus	400046	Single	ReadWrite	WordSwap=True
Tempo_AT	modbus	400048	Single	Read	WordSwap=True
Degrau_AT	modbus	400050	Single	ReadWrite	WordSwap=True

Os testes confirmaram que a comunicação apresentou desempenho estável e com atu-

alização rápida, permitindo acompanhar o nível do tanque e as respostas dos controladores. As funcionalidades de ativação de degrau, alteração de parâmetros e coleta de dados funcionaram conforme o esperado, demonstrando que a integração entre o supervisório e o CLP foi bem-sucedida e adequada ao propósito da plataforma de controle desenvolvida.

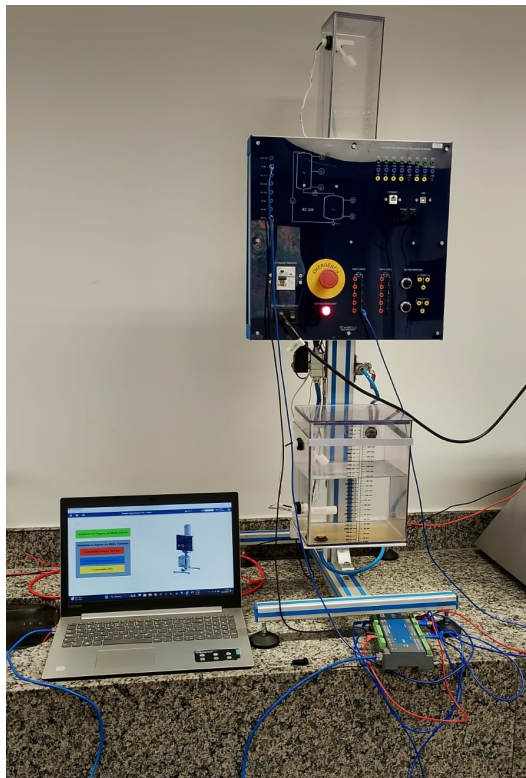
## 5.6 Configuração da Plataforma na Bancada de Ensaios

Após a conclusão das etapas de desenvolvimento das telas do sistema supervisório e da configuração da comunicação entre o BluePlant e o CLP Altus, deu-se início ao processo de configuração da bancada didática. Com todas as *tags* do supervisório devidamente vinculadas às variáveis internas do CLP, tornou-se possível validar fisicamente a plataforma e realizar os testes experimentais planejados.

Como a bancada física já se encontrava construída e em funcionamento, esta etapa consistiu apenas na conexão do CLP às entradas e saídas correspondentes à bomba e ao transmissor de nível, além da conexão do CLP ao computador via cabo Ethernet, utilizado para operar o software supervisório BluePlant.

A Figura 16 apresenta a configuração final da Plataforma, juntamente com a bancada, mostrando a estrutura e os componentes utilizados durante os experimentos.

Figura 16 – Configuração da plataforma na bancada de ensaios.



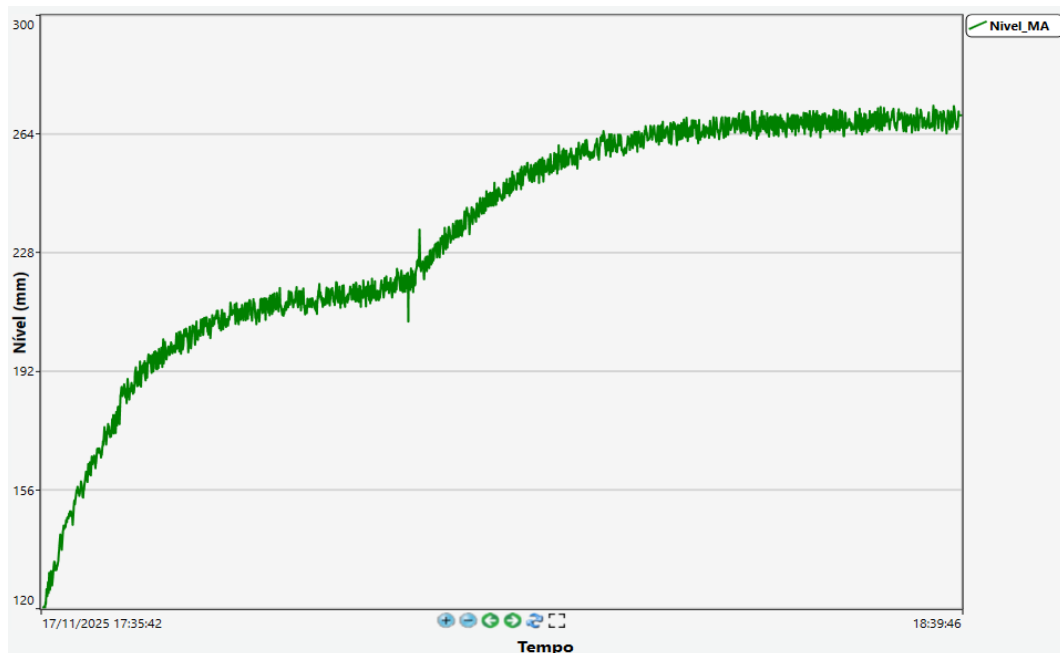
## 5.7 Obtenção da Resposta em Malha Aberta

Com a plataforma integrada ao CLP e ao sistema supervisorio, foi realizada a obtenção da resposta em malha aberta para uma entrada em degrau unitário do sistema de tanques. Para isso, a bomba foi inicialmente acionada com um valor de referência igual a 17000 unidades digitais do CLP, correspondente a aproximadamente 13,6 V na saída PWM, permitindo que o nível do tanque atingisse o regime permanente em um valor diferente de zero. Para esse valor inicial aplicado, o sistema obteve um valor final de aproximadamente 220 mm.

Após atingir o estado estacionário, o valor aplicado à bomba foi elevado para 25000 unidades digitais do CLP, correspondente a aproximadamente 20,0 V na saída PWM, caracterizando um degrau de entrada no sistema. Novamente, aguardou-se até que o nível atingisse o estado estacionário, sendo obtido ao final o valor de aproximadamente 271 mm.

A Figura 17 apresenta o gráfico do comportamento do sistema em resposta ao degrau em malha aberta, exibido na tela do supervisorio da plataforma.

Figura 17 – Gráfico da resposta em malha aberta do sistema apresentada pelo supervisorio.

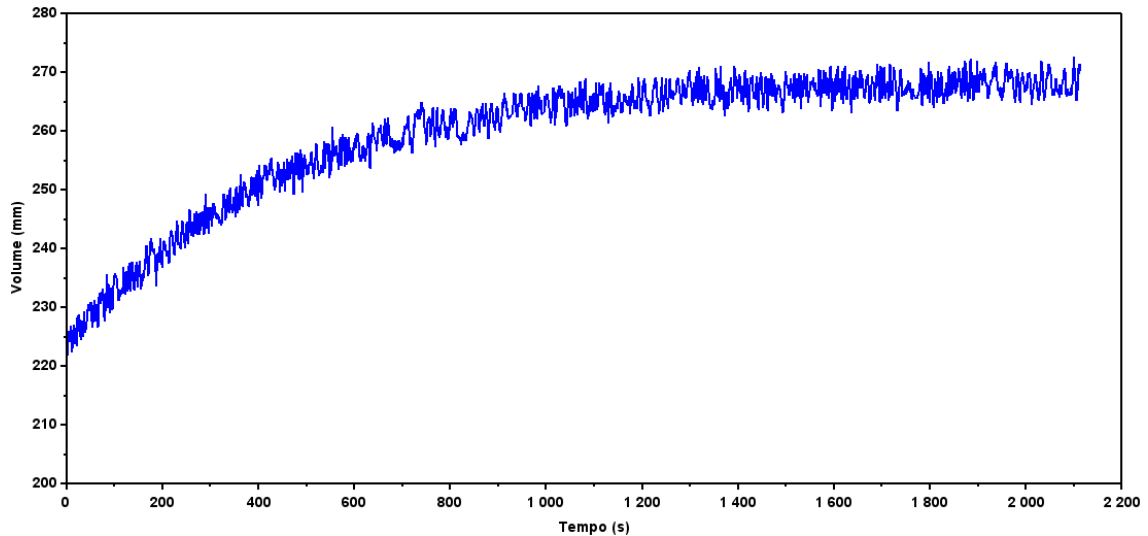


Fonte: Próprio autor.

Após obter a resposta em malha aberta, foi utilizada a opção de salvar os dados registrados em uma planilha no formato Excel. Em seguida, os dados do arquivo exportado foram carregados no Scilab, permitindo a visualização da curva com maior precisão. A Figura 18 apresenta a curva de resposta ao degrau em malha aberta obtida no Scilab, considerando apenas os dados adquiridos após o sistema ter atingido o regime permanente inicial. Em relação à Figura 17, os dados correspondentes à fase de subida inicial foram desconsiderados, de modo que a análise

passa a se concentrar exclusivamente no trecho em que foi aplicado o degrau de entrada, com a elevação do valor de acionamento da bomba para 25000.

Figura 18 – Degrau em Malha Aberta no Scilab.



Fonte: Próprio autor.

A partir dos valores inicial e final da entrada aplicada à bomba e do nível observado no gráfico, foi possível estimar o ganho estático  $K$ , que é utilizado na elaboração da função de transferência aproximada do sistema, através da função:

$$K = \frac{V_f - V_i}{U_f - U_i}, \quad (5.9)$$

em que  $V_i$  e  $V_f$  representam, respectivamente, os valores inicial e final do nível do tanque (em milímetros), e  $U_i$  e  $U_f$  correspondem aos valores inicial e final do sinal de entrada aplicado à bomba. Substituindo os valores obtidos na Eq. (5.9), tem-se:

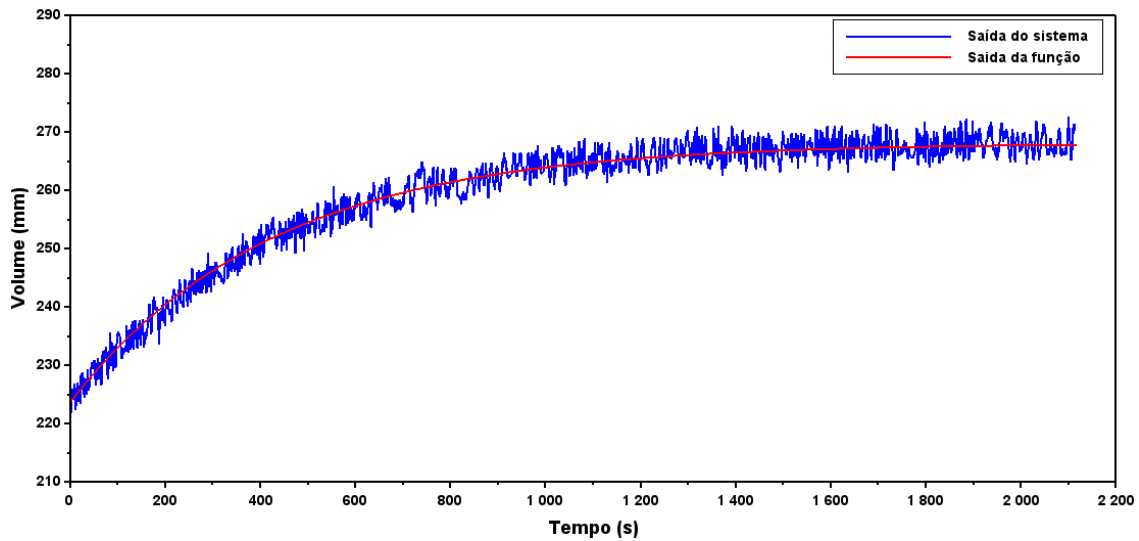
$$K = \frac{271 - 220}{25000 - 17000} = 0,00638$$

Além disso, estimou-se a constante de tempo  $T$  do sistema a partir do instante em que a resposta atingiu 63,2% do valor final, obtendo-se:

$$T = 423,75 \text{ segundos.}$$

Para confirmação dos valores obtidos para o ganho estático e a constante de tempo, foi gerado novamente no Scilab o gráfico da resposta ao degrau, desta vez acompanhado da curva correspondente à função gerada a partir dos parâmetros obtidos. Essa comparação permitiu validar visualmente os valores identificados e reforçar a precisão dos cálculos realizados. A Figura 19 apresenta a curva de resposta em degrau com a resposta da função com ganho ajustado como referência.

Figura 19 – Degrau em Malha Aberta com a linha de resposta da função.



Fonte: Próprio autor.

A partir da confirmação dos valores de ganho estático e constante de tempo por meio do gráfico da resposta ao degrau com a linha de referência no Scilab, foi gerada a função de transferência em malha aberta do sistema:

$$G_p(s) = \frac{0,00638}{423,75s + 1}. \quad (5.10)$$

Essa função descreve exclusivamente a dinâmica do sistema físico de tanques, sem a inclusão de ações de controle. O termo  $\frac{1}{s}$ , correspondente à ação integral, não faz parte do modelo da planta, sendo introduzido posteriormente como parte do controlador, com o objetivo de eliminar o erro em regime permanente para uma entrada do tipo degrau.

Assim, a função de transferência total em malha aberta utilizada no projeto do controlador é dada por:

$$G(s) = G_p(s) G_c(s) = \left( \frac{0,00638}{423,75s + 1} \right) \frac{1}{s}$$

que pode ser reescrita como:

$$G(s) = \frac{0,0000151}{s(s + 0,00236)}$$

A partir desse modelo, a função de transferência em malha fechada do sistema com a ação integral foi determinada como:

$$\frac{C(s)}{R(s)} = \frac{0,0000151}{s^2 + 0,00236s + 0,0000151}$$

Para essa função de transferência, identificaram-se os pólos dominantes em  $s = -0,00118 \pm j0,000037$ . Comparando a função de transferência obtida com a forma padrão de um sistema de

segunda ordem apresentada na Eq. (3.3), tem-se:

$$\omega_n^2 = 0,0000151 \Rightarrow \omega_n = \sqrt{0,0000151} \approx 0,003886,$$

$$2\zeta\omega_n = 0,00236 \Rightarrow \zeta = \frac{0,00236}{2\omega_n} \approx 0,30365.$$

A partir desses parâmetros, o sobressinal máximo foi estimado pela expressão apresentada na Eq. (3.5), resultando em:

$$M_p \approx 36,74\%.$$

O tempo de acomodação (critério de 2%) foi estimado pela expressão apresentada na Eq. (3.6), resultando em um tempo de acomodação para a resposta em malha fechada de:

$$t_s \approx \frac{4}{0,30365 \cdot 0,003886} \approx 3389,9 \text{ s.}$$

## 5.8 Controlador de Avanço de Fase

Após a análise da função de transferência do sistema em malha fechada, verificou-se que os valores de sobressinal e de tempo de acomodação são significativamente elevados, resultando em uma resposta lenta e com oscilação significativa. Esse comportamento evidencia a necessidade de aplicar técnicas de compensação para aprimorar o desempenho dinâmico do processo. Seguindo o planejamento estabelecido para o trabalho, o primeiro controlador projetado foi o controlador de avanço de fase, cuja função é antecipar a resposta do sistema, aumentando sua velocidade e reduzindo o sobressinal por meio do deslocamento dos pólos para regiões mais adequadas no plano- $s$ .

### 5.8.1 Cálculo dos Parâmetros do Controlador

O projeto do controlador foi realizado a partir da função de transferência em malha aberta identificada na etapa anterior, obtida por meio dos dados registrados pelo sistema supervisório.

Foram definidos como objetivos para o sistema compensado a redução do sobressinal para aproximadamente 10% e um tempo de acomodação próximo de 480 segundos. Para atender a essas especificações, foram calculados inicialmente o fator de amortecimento  $\zeta$  e a frequência natural  $\omega_n$  da malha fechada desejada.

Para um sobressinal máximo especificado de 10%, utilizou-se a relação apresentada na Eq. (3.5), da qual se isolou o fator de amortecimento  $\zeta$ , obtendo-se:

$$\zeta = 0,5912.$$

Em seguida, a partir do tempo de acomodação desejado e da Eq. (3.6), isolou-se a frequência natural  $\omega_n$ , resultando em:

$$\omega_n = \frac{4\zeta}{T_s} = \frac{4 \cdot 0,5912}{480} = 0,0145.$$

Assim, os pólos dominantes desejados para o sistema compensado tornam-se:

$$s_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} = -0,0086 \pm j0,0117.$$

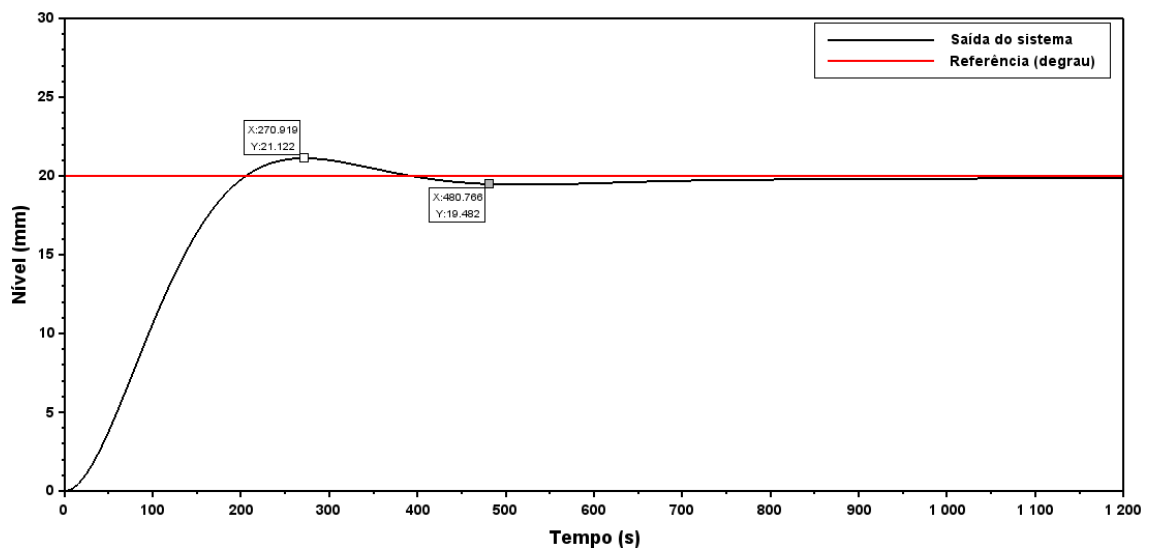
Com base nesses valores e comparando a dinâmica desejada com a dinâmica original do processo, determinou-se o controlador de avanço pelo Método do Lugar das Raízes, resultando na seguinte forma:

$$G_c(s) = 13,717 \cdot \frac{s + 0,00183}{s + 0,0172}.$$

### 5.8.2 Simulação no Scilab

O controlador projetado, foi aplicado ao modelo do processo e seu desempenho foi avaliado por meio de simulações no *software* Scilab, utilizando um degrau de amplitude 20 mm, valor que também será aplicado nos ensaios do controlador. A Figura 20 apresenta a resposta ao degrau do sistema compensado.

Figura 20 – Resposta ao degrau da simulação do sistema com o controlador de avanço de fase.



Fonte: Próprio autor.

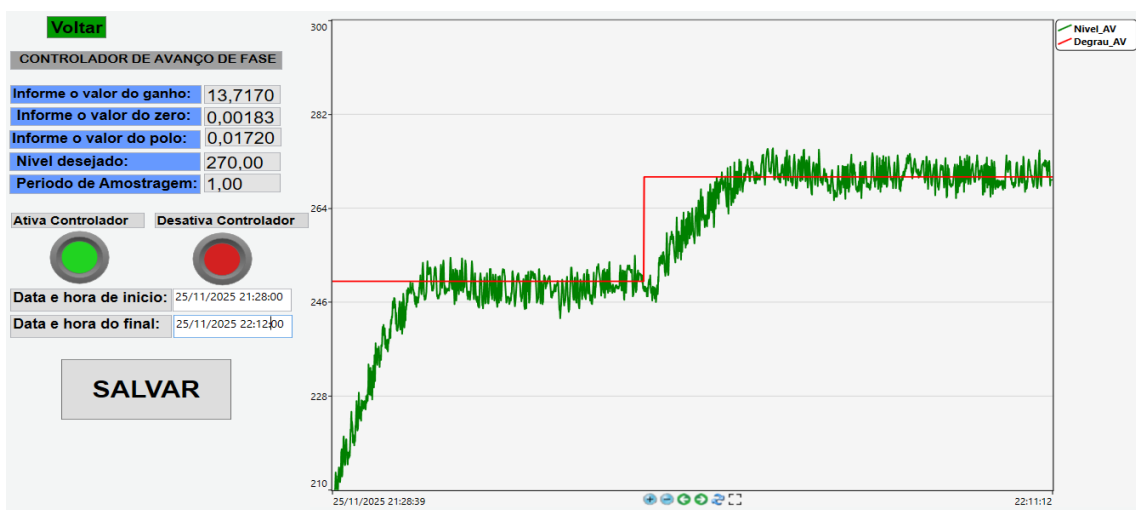
A simulação da resposta ao degrau apresentou desempenho compatível com os requisitos especificados: o sobressinal foi reduzido para valores inferiores ao limite estabelecido, apresentando um valor de aproximadamente 5%, e o tempo de acomodação atingiu o valor desejado de 480 s. Esses resultados indicam que o controlador de avanço antecipou eficazmente a dinâmica do processo, melhorando a velocidade de resposta sem comprometer a estabilidade do sistema.

### 5.8.3 Resultado Prático no Supervisório

Após a validação na simulação, o controlador foi implementado no CLP para avaliação prática. Os parâmetros: polo, zero, ganho do controlador e tempo de amostragem foram configurados diretamente na tela do supervisório. O tempo de amostragem utilizado foi de 1 segundo.

Durante o ensaio, aplicou-se um degrau de 250 mm e, após o regime permanente ser alcançado, o valor desejado foi ajustado para 270 mm. O supervisório registrou a evolução do nível em malha fechada, permitindo observar a atuação do controlador em tempo real.

Figura 21 – Tela do supervisório com parâmetros inseridos e resposta prática do sistema.

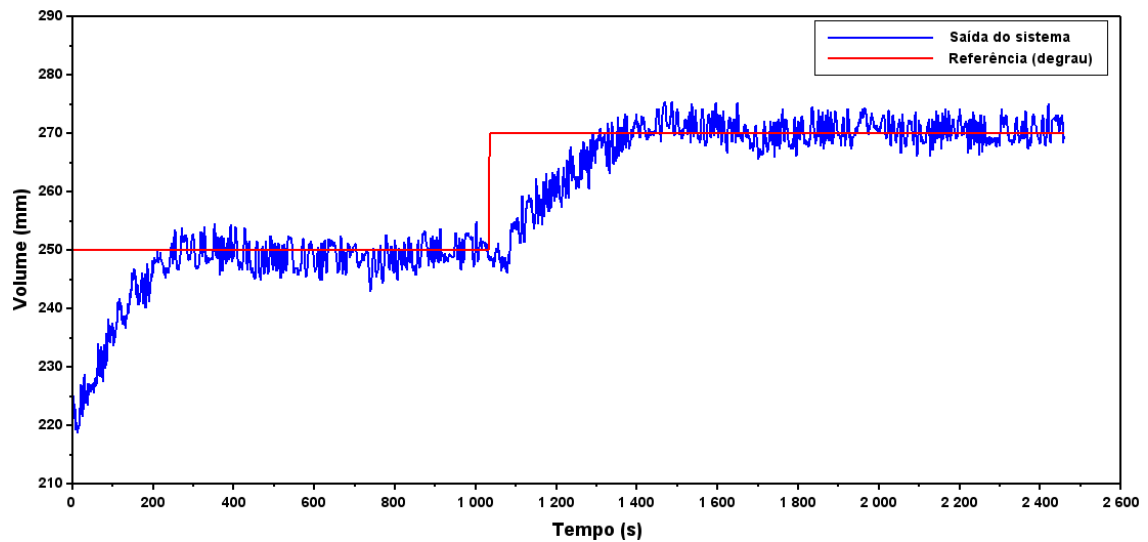


Fonte: Próprio autor.

A Figura 21 permite visualizar como o sistema se comporta quando o controlador de avanço está atuando. Na tela do supervisório, é possível observar os valores do controlador que foram configurados e acompanhar, em tempo real, como o nível do tanque responde às mudanças solicitadas, primeiro para 250 mm e depois para 270 mm. Observa-se que o tanque atinge os valores desejados e permanece estável em ambos os valores de degrau aplicados, sem oscilações excessivas e atendendo ao tempo de acomodação e sobressinal estabelecidos. Isso mostra que o controlador está funcionando corretamente, ajudando o sistema a atingir o nível desejado, com um valor de sobressinal baixo e um tempo de acomodação próximo ao desejado.

Os dados registrados durante o ensaio foram exportados em formato de planilha e tratados no Scilab para análise detalhada. A Figura 22 apresenta o gráfico reconstruído da resposta ao degrau.

Figura 22 – Resposta ao degrau obtida no Scilab a partir dos dados exportados.

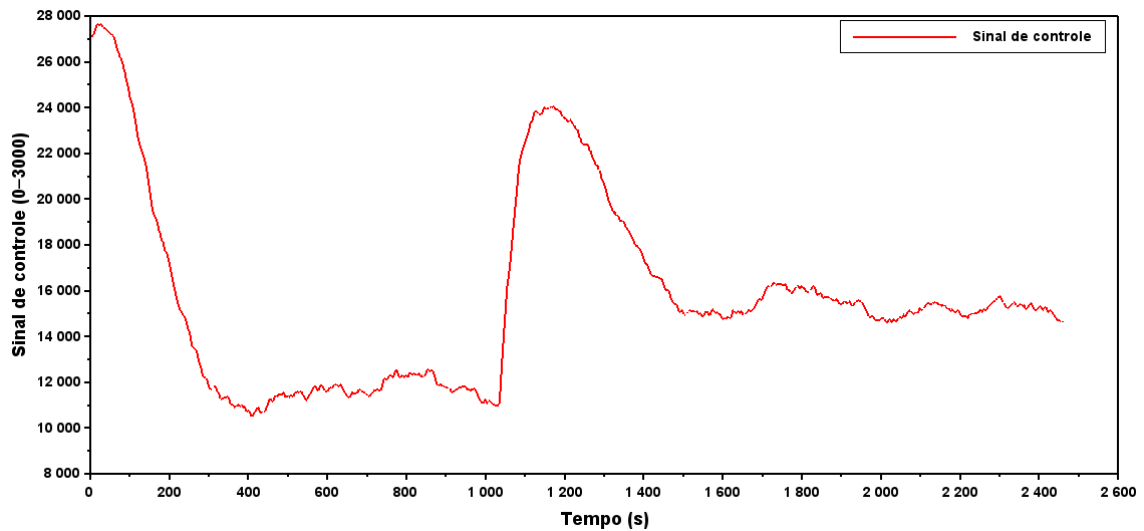


Fonte: Próprio autor.

A análise do gráfico permite observar de forma mais clara que os dados obtidos no ensaio prático seguem a tendência prevista na resposta ao degrau da simulação apresentada na Figura 20. Observa-se que o nível do tanque atinge o valor desejado dentro do tempo de acomodação especificado e sem ultrapassar significativamente o valor de referência. O sobressinal permaneceu abaixo do limite estabelecido no projeto, apresentando também no ensaio prático um valor de aproximadamente 5%, indicando uma atuação adequada do controlador. Dessa forma, a curva confirma que o sistema respondeu de maneira estável, previsível e de acordo com os critérios de desempenho definidos.

Também foi analisado o sinal de controle aplicado à bomba. A Figura 23 apresenta o sinal de controle escalonado pelo CLP (0–3000), proporcional à tensão de 0 a 24 V, ao longo da execução do algoritmo de compensação.

Figura 23 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador de Avanço de Fase.



Fonte: Próprio autor.

A análise do sinal de controle aplicado à bomba permite observar que no início do ensaio, a bomba opera com um valor de tensão elevado, o que é esperado, pois o controlador aumenta o esforço de atuação para acelerar a subida do nível até o valor desejado. À medida que o sistema se aproxima da referência, o sinal de controle aplicado diminui gradualmente, estabilizando-se em um patamar compatível com a manutenção do nível. O sinal permanece sem variações significativas, indicando que o controlador ajustou corretamente a atuação da bomba e manteve o sistema em regime permanente com estabilidade e sem oscilações indesejadas.

Os resultados obtidos confirmam que a plataforma apresentou um desempenho satisfatório na realização do ensaio em malha fechada com o controlador de avanço. Tanto a resposta em degrau registrada pelo supervisor quanto a análise dos dados exportados no Scilab demonstraram coerência com o comportamento esperado, atendendo aos requisitos de tempo de acomodação e apresentando sobressinal inferior ao limite estabelecido. Além disso, o sinal de atuação da bomba se manteve estável e compatível com a dinâmica do processo, evidenciando que o controlador operou de forma eficiente em ambiente real. Dessa forma, a resposta da plataforma ao degrau validou tanto o modelo quanto o projeto do compensador, reforçando a confiabilidade do sistema desenvolvido.

## 5.9 Controlador PI (implementação a partir do PID)

Embora a proposta inicial incluísse o estudo de um controlador PID, o modelo obtido para o sistema da plataforma de tanques acoplados apresenta comportamento equivalente a um processo de primeira ordem dominante, com dinâmica lenta. Em sistemas com essa dinâmica, a ação derivativa tende a não trazer benefícios significativos para o desempenho e ainda pode

amplificar ruídos na resposta do sistema, tornando sua utilização desnecessária. Dessa forma, optou-se por considerar apenas as ações proporcional e integral, resultando na implementação de um controlador do tipo PI. Essa escolha mantém a simplicidade do projeto, evita efeitos indesejados no atuador e preserva a capacidade de correção do erro em regime permanente, atendendo adequadamente às necessidades de controle do processo.

### 5.9.1 Cálculo dos Parâmetros do Controlador

O controlador PI foi projetado com base no método do posicionamento de pólos em malha fechada, a partir do modelo aproximado de primeira ordem da planta obtido na etapa de identificação. Nesse método, os parâmetros do controlador são determinados de forma a fazer com que os pólos dominantes da função de transferência em malha fechada coincidam com os pólos desejados, calculados a partir das especificações de desempenho.

Como requisitos de desempenho, foram estabelecidos um sobressinal máximo de aproximadamente 12% e um tempo de acomodação em torno de 600 segundos. A partir dessas especificações, determinaram-se inicialmente o fator de amortecimento  $\zeta$  e a frequência natural  $\omega_n$  da dinâmica desejada para o sistema em malha fechada.

Para um sobressinal máximo de 12%, utilizou-se a relação apresentada na Eq. (3.5), resultando em:

$$\zeta = 0,5444.$$

Em seguida, a partir do tempo de acomodação desejado e da relação apresentada na Eq. (3.6), isolou-se a frequência natural, resultando em:

$$t_s = \frac{4}{\zeta\omega_n} \Rightarrow \omega_n = \frac{4}{600 \cdot 0,5444} = 0,0122.$$

O controlador PI contínuo é descrito por (OGATA, 2010):

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} \right), \quad (5.11)$$

onde  $K_p$  representa o ganho proporcional do controlador PI e  $T_i$  é a constante de tempo da ação integral, responsável por eliminar o erro em regime permanente para entradas do tipo degrau.

Aplicando o método do posicionamento de pólos à função de transferência em malha fechada e igualando o polinômio característico ao polinômio desejado de segunda ordem,

$$s^2 + 2\zeta\omega_n s + \omega_n^2, \quad (5.12)$$

obtêm-se as expressões para os parâmetros do controlador (OGATA, 2010):

$$K_p = \frac{2\zeta\omega_n T - 1}{k}, \quad (5.13)$$

$$T_i = \frac{2\zeta\omega_n T - 1}{\omega_n^2 T}. \quad (5.14)$$

Substituindo-se os valores  $\zeta = 0,5444$ ,  $\omega_n = 0,0122$ ,  $T = 423,75$  e  $k = 0,00639$ , obtêm-se:

$$K_p = 724,39,$$

$$T_i = 73,39.$$

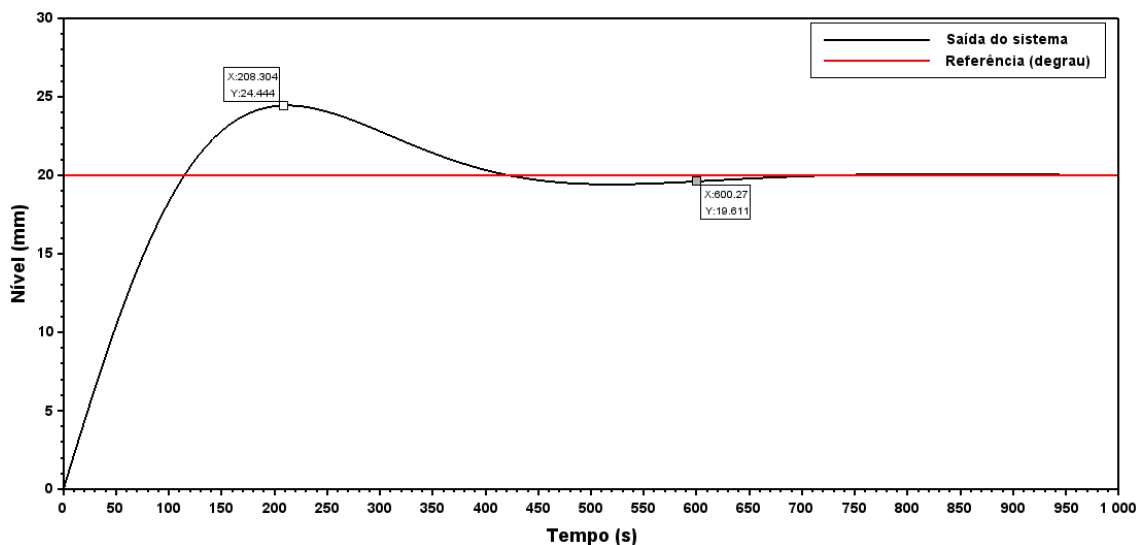
Com os valores obtidos, a função de transferência do controlador PI é dada por:

$$G_c(s) = 724,39 \left( 1 + \frac{1}{73,39s} \right) = 724,39 \left( \frac{73,39s + 1}{73,39s} \right).$$

### 5.9.2 Simulação no Scilab

Com o controlador projetado, o mesmo foi aplicado ao modelo do processo e avaliado por meio de simulação no *software* Scilab, utilizando um degrau de amplitude 20 mm, valor que também será aplicado nos ensaios do controlador. A Figura 24 apresenta a resposta ao degrau do sistema compensado.

Figura 24 – Resposta ao degrau da simulação do sistema com o controlador PI.



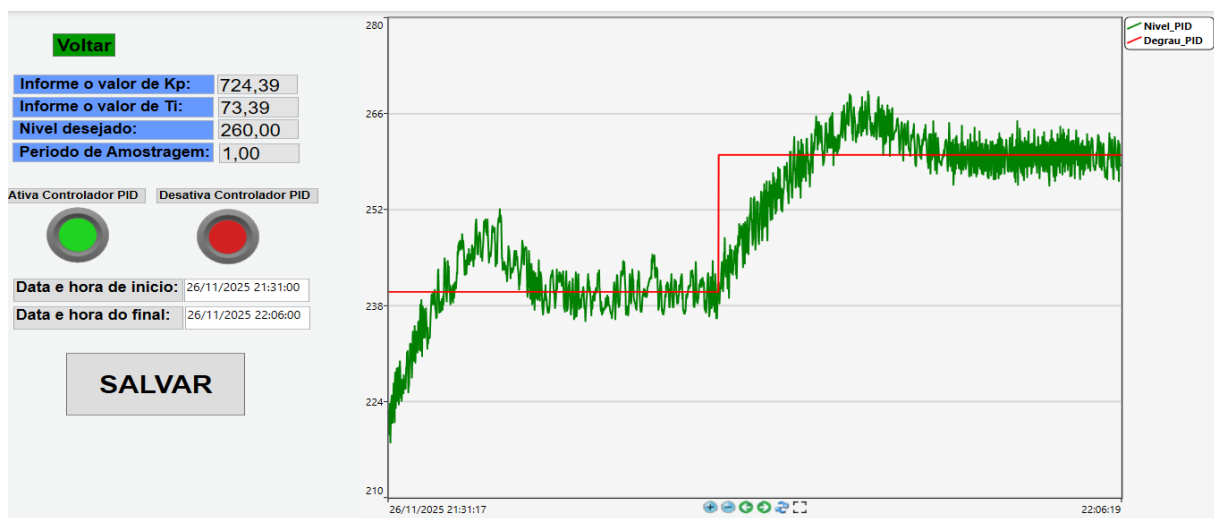
Fonte: Próprio autor.

Na simulação do controlador PI, observou-se que o tempo de acomodação atendeu ao valor especificado de 600 segundos, mas o sobressinal ultrapassou o valor esperado, apresentando um valor de aproximadamente 22%. Esse comportamento é comum, pois a ação integral do PI tende a tentar eliminar o erro de forma imediata, acumulando integrador no início da resposta. Em um processo de primeira ordem, essa ação torna a resposta equivalente à de um sistema de segunda ordem com menor amortecimento, aumentando naturalmente o sobressinal. Assim, mesmo com o sobressinal acima do previsto, o controlador está correto, pois os valores de  $K_p$  e  $T_i$  foram obtidos de acordo com o método de projeto adotado. A diferença observada é consequência natural da dinâmica real do sistema e não de erro no cálculo do PI.

### 5.9.3 Implementação e Resultado Prático no Supervisório

Seguindo o mesmo procedimento anterior, o controlador PI foi implementado no CLP. Os parâmetros calculados, ganho proporcional  $K_p$  e tempo integrador  $T_i$ , foram inseridos na interface do supervisório, e o ensaio prático foi realizado com período de amostragem de 1 s. Durante o experimento, aplicou-se um degrau de 240 mm e, após a estabilização, o valor desejado foi ajustado para 260 mm. A Figura 25 ilustra os resultados obtidos ao experimento do sistema com o controlador PI.

Figura 25 – Tela do supervisório com parâmetros inseridos e resposta prática do sistema com o controlador PI.

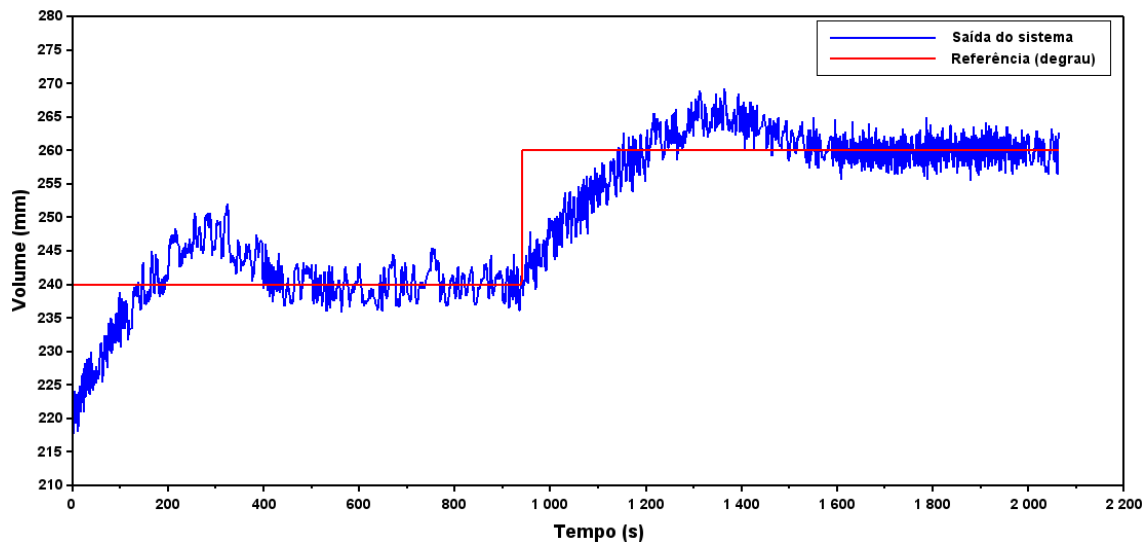


Fonte: Próprio autor.

A resposta obtida no supervisório durante o ensaio prático apresentou comportamento próximo ao previsto na simulação, evidenciando a boa correspondência entre o modelo e o processo real. Assim como observado no ambiente simulado, o sistema apresentou um sobressinal acima do especificado, consequência natural da ação integral ao tentar eliminar o erro rapidamente. No entanto, o tempo de acomodação foi atendido nos dois níveis de referência, demonstrando que o controlador PI foi capaz de rastrear a referência do nível de forma eficiente. A curva registrada no supervisório confirma que, apesar do pico inicial, o sistema convergiu de maneira adequada e o controlador manteve o nível no valor desejado, validando o desempenho do PI implementado no CLP.

Os dados registrados durante o ensaio também foram exportados em formato de planilha e tratados no Scilab para análise detalhada. A Figura 26 apresenta o gráfico reconstruído da resposta ao degrau.

Figura 26 – Resposta ao degrau obtida no Scilab a partir dos dados exportados do controlador PI.

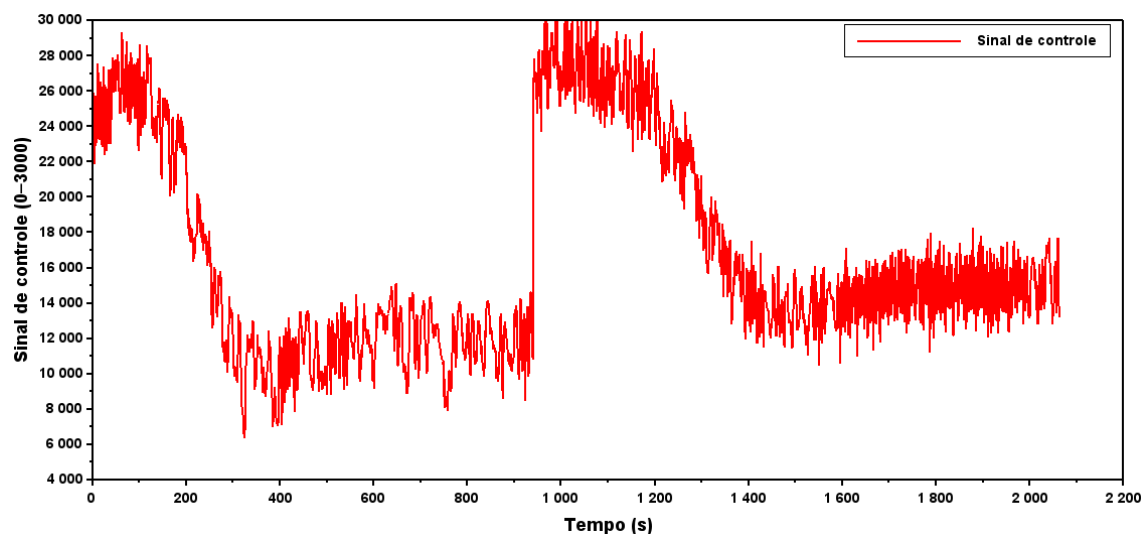


Fonte: Próprio autor.

O gráfico da simulação no Scilab apresentou um comportamento similar ao verificado no supervisor, confirmando a fidelidade do modelo utilizado. Assim como no experimento real, houve um sobressinal acima do desejado, porém o tempo de acomodação foi atendido. Essa correspondência entre simulação e supervisor destaca o êxito do controlador PI e da plataforma, demonstrando que os parâmetros calculados foram adequados e que o sistema se comportou conforme o previsto nas duas etapas.

Também foi analisado o sinal de atuação aplicado à bomba. A Figura 27 apresenta o sinal de controle ao longo da execução do algoritmo de compensação.

Figura 27 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador PI.



Fonte: Próprio autor.

A análise do sinal de controle aplicado à bomba, exportado do ensaio realizado no supervísório apresentado na Figura 25, mostra que, com o controlador PI, o sinal de controle da bomba apresentou oscilações significativamente maiores em comparação com o desempenho da bomba na aplicação do controlador de avanço de fase. O maior nível de oscilações no sinal de controle com o controlador PI está associado principalmente à menor margem de fase do sistema nessa configuração, uma vez que ambos os algoritmos possuem ação integral, e o compensador por avanço de fase proporciona maior amortecimento da resposta, resultando em um sinal de controle mais suave e com menor nível de oscilações. Ressalta-se que o controlador PI foi intencionalmente projetado para apresentar um sobressinal maior do que o compensador por avanço de fase, conforme os critérios adotados no projeto.

Observa-se que o sinal iniciou com amplitude elevada durante a transição e gradualmente foi se atenuando à medida que o nível aproximou-se do valor de referência, passando a oscilar em torno de um valor quase constante quando o nível entrou em regime permanente. Apesar do nível de oscilações maior, o sistema acompanhou as mudanças de referência e atingiu o tempo de acomodação desejado, confirmando o êxito do controle.

De forma geral, os resultados obtidos demonstram o êxito da plataforma em reproduzir a resposta em degrau utilizando o controlador PI. Tanto a simulação quanto o supervísório apresentaram comportamentos coerentes entre si, confirmando a validade do modelo adotado e a correta implementação dos parâmetros do controlador no CLP. Apesar das oscilações características da ação integral, o sistema foi capaz de acompanhar as variações de referência e estabilizar o nível dentro do tempo previsto, comprovando que a plataforma atendeu plenamente aos objetivos propostos para o controle PI.

## 5.10 Controlador de Atraso de Fase

Diferente do controlador de avanço de fase, o controlador de atraso não tem como principal função melhorar a resposta transitória do sistema, mas sim alterar suas características em regime permanente por meio da modificação do ganho estático. No caso do sistema da bancada de ensaios utilizada no projeto, que é caracterizado como um sistema de primeira ordem, verificou-se que o cálculo do controlador a partir da resposta ao degrau não conduziria a um projeto eficaz, uma vez que o sistema já apresenta erro estacionário nulo para entradas do tipo degrau.

Dessa forma, o projeto do controlador de atraso de fase foi realizado a partir da resposta do sistema a uma entrada do tipo rampa. Essa escolha permitiu avaliar o desempenho do sistema em relação ao erro de regime permanente, caracterizado pela constante de erro de velocidade ( $K_v$ ). A partir desse critério, foi possível dimensionar o compensador de atraso de fase com o objetivo principal de melhorar o desempenho em regime permanente. Após o projeto, o controlador foi então aplicado ao sistema e sua resposta ao degrau foi analisada, permitindo a avaliação do comportamento transitório da malha compensada.

Para o projeto do controlador de atraso de fase, foi realizado novamente o processo de modelagem do sistema a partir de um novo ensaio de resposta em malha aberta. Essa nova identificação tornou-se necessária porque, em função do uso prático da plataforma por outros usuários, as configurações físicas do sistema foram alteradas, especialmente a posição da válvula manual que controla a vazão de saída do tanque. Essas modificações interferem diretamente na dinâmica do processo, alterando os parâmetros do modelo matemático. Dessa forma, a obtenção de uma nova função de transferência garantiu maior coerência entre o modelo e o comportamento real da planta nas condições específicas de operação adotadas para esse controlador. A identificação seguiu as mesmas etapas previamente descritas neste trabalho, envolvendo a coleta de dados via supervisor, a exportação dos sinais e o tratamento das informações no Scilab.

Ao final desse processo, obteve-se uma nova função de transferência aproximada para o sistema em malha aberta, dada por:

$$G(s) = \frac{0,00000483}{(s + 0,00222)}.$$

Considerando a inclusão da ação integral  $\frac{1}{s}$  no controlador, a função de transferência completa do sistema em malha aberta passa a ser dada por:

$$G(s) = G_c(s) G_p(s) = \frac{1}{s} \cdot \left( \frac{0,00000483}{s + 0,00222} \right).$$

### 5.10.1 Cálculo dos Parâmetros do Controlador

Para a elaboração do controlador de atraso de fase utilizando o método baseado na resposta em rampa, inicialmente foi necessário determinar o desempenho do sistema original em relação ao erro de regime permanente para esse tipo de entrada. Para isso, utilizou-se a constante de erro de velocidade ( $K_v$ ), definida para sistemas cuja função de transferência em malha aberta é dada por  $G(s)$ .

A constante  $K_v$  é calculada como:

$$K_v = \lim_{s \rightarrow 0} s G(s).$$

Substituindo a função modelada experimentalmente,

$$G(s) = \frac{0,00000483}{s(s + 0,00222)},$$

obtem-se:

$$K_v = \lim_{s \rightarrow 0} s \cdot \frac{0,00000483}{s(s + 0,00222)} = \lim_{s \rightarrow 0} \frac{0,00000483}{s + 0,00222} = \frac{0,00000483}{0,00222}.$$

Portanto,

$$K_v = 0,00218.$$

A partir de  $K_v$ , o erro de regime permanente para entrada rampa unitária é dado por:

$$e_{ss} = \frac{1}{K_v},$$

resultando em:

$$e_{ss} = \frac{1}{0,00218} = 458,72.$$

O elevado valor de erro em regime permanente demonstra que o sistema, sem compensação, não apresenta bom desempenho para resposta ao sistema em rampa. Dessa forma, torna-se necessária a implementação de um controlador de atraso de fase, com o objetivo de aumentar a constante de erro de velocidade e reduzir o erro estacionário. Para isso, definiu-se como meta limitar o erro final a um valor máximo de:

$$e_{ss,desejado} = 100.$$

Com o novo requisito, a constante de erro de velocidade desejada passa a ser:

$$K_{v,desejado} = \frac{1}{e_{ss,desejado}} = \frac{1}{100} = 0,01.$$

Para atender ao erro estacionário desejado, definiu-se a constante de erro de velocidade necessária como:

$$K_{v,d} = \frac{1}{e_{ss,desejado}} = \frac{1}{100} = 0,01.$$

Assim, o fator de atraso pode ser obtido pela razão entre o valor desejado e o valor original da constante de erro de velocidade:

$$B = \frac{K_{v,d}}{K_v} = \frac{0,01}{0,00218} = 4,59.$$

Como o compensador de atraso possui a forma:

$$G_c(s) = K_c \frac{s+z}{s+p},$$

O fator  $B$  também pode ser interpretado diretamente a partir da própria estrutura do compensador, pois representa a razão entre o zero e o polo do controlador de atraso. Dessa forma, uma vez conhecido o valor de  $B$ , define-se inicialmente a posição do zero desejado e, a partir dele, determina-se o polo correspondente dividindo o zero pelo fator  $B$ . No projeto realizado, adotou-se o zero em  $z = 0,0001$  e, ao dividi-lo pelo valor de  $B = 4,59$ , obteve-se o polo do compensador, que resulta aproximadamente em  $p = 0,0000218$ . O ganho  $K_c$  foi obtido seguindo o procedimento do Método do Lugar das Raízes, de forma a garantir que o sistema compensado atendesse aos requisitos dinâmicos desejados sem comprometer a estabilidade. A partir da análise do lugar das raízes da função de transferência compensada, ajustou-se o ganho até que os polos dominantes assumissem a localização apropriada no plano  $s$ , resultando no valor final de  $K_c = 1,0204$ .

A partir dos parâmetros definidos, ganho  $K_c$ , zero e polo do compensador, foi possível estabelecer a forma final do controlador de atraso de fase, resultando na seguinte função de transferência:

$$G_c(s) = 1,0204 \cdot \frac{s + 0,0001}{s + 0,0000218}.$$

Para o controlador de atraso de fase, a nova constante de erro de velocidade é obtida pela relação:

$$K_v f = K_c \cdot B \cdot K_v$$

onde  $K_c = 1,0204$ ,  $B = 4,59$  e  $K_v$  é a constante original do sistema. Substituindo os valores, obtém-se:

$$K_v f \approx 0,0104.$$

Assim, o erro estacionário para uma entrada rampa unitária passa a ser:

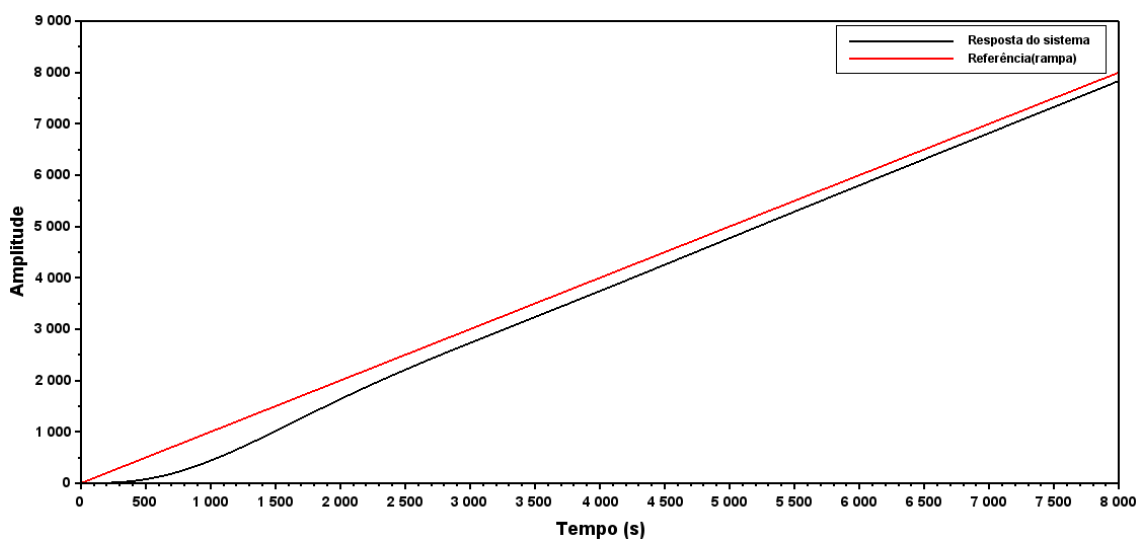
$$e_{ss} = \frac{1}{K_v f} \approx 96,0.$$

Dessa forma, o controlador aumentou a constante de erro de velocidade do processo, reduzindo o erro estacionário em relação ao sistema não compensado.

### 5.10.2 Simulação no Scilab

Antes de analisar a resposta do sistema ao degrau de referência com o controlador implementado, foi analisado o comportamento da planta em relação a uma entrada em rampa, a fim de verificar a sua capacidade de seguimento da referência. A Figura 28 apresenta a resposta do sistema à rampa aplicada no supervisor.

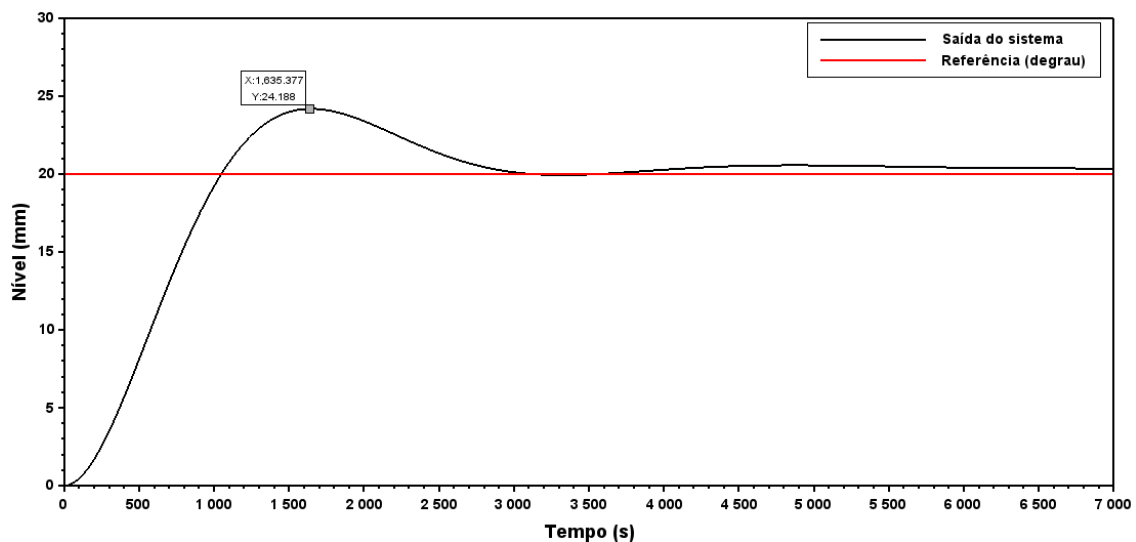
Figura 28 – Resposta a simulação do sistema à entrada em rampa.



A resposta do sistema à entrada em rampa mostra que a saída acompanha a tendência crescente do sinal de referência com inclinação praticamente paralela, indicando boa capacidade de seguimento da forma do sinal. Observa-se que o erro de acompanhamento é maior no início da resposta e vai sendo gradualmente reduzido, até se estabilizar em um valor praticamente constante. A partir do gráfico, estima-se que esse erro final permanece próximo de 100, valor coerente com o resultado desejado para a resposta em rampa.

A aplicação do controlador de atraso de fase foi iniciada pela etapa de simulação no Scilab, utilizando o modelo compensado obtido a partir do projeto apresentado anteriormente. Embora o controlador tenha sido desenvolvido com foco na redução do erro estacionário para entrada em rampa, buscou-se verificar também seu comportamento frente a uma entrada em degrau, avaliando aspectos como estabilidade, tempo de acomodação e suavidade da resposta. A Figura 29 apresenta a resposta ao degrau do sistema compensado pela ação do controlador de atraso.

Figura 29 – Resposta ao degrau da simulação do sistema com o controlador de Atraso de Fase.



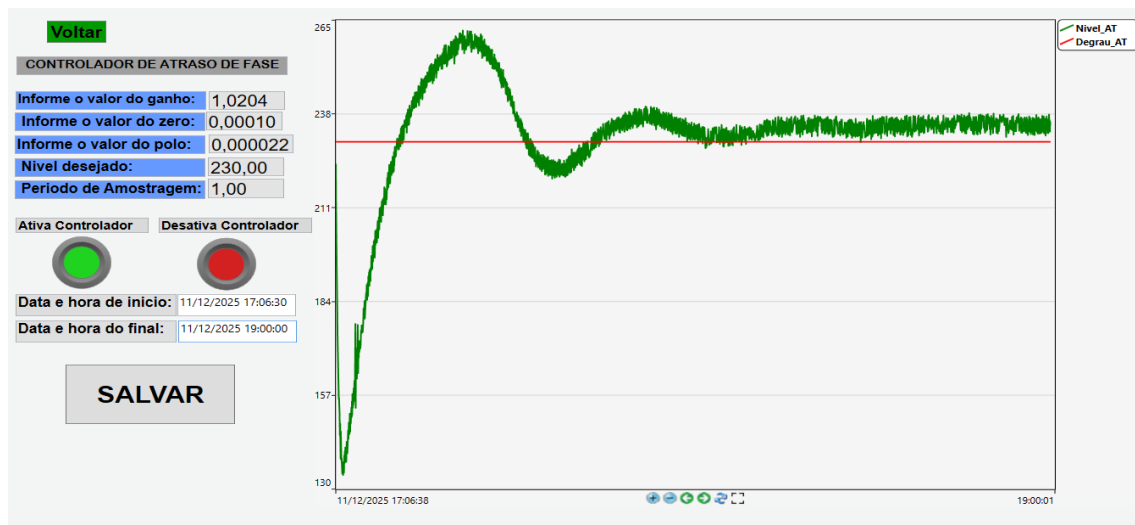
Fonte: Próprio autor.

Ao analisar o gráfico de resposta, observa-se um comportamento transitório lento, seguido de um sobressinal expressivo, atingindo aproximadamente 22%, e um decaimento gradual que resulta em uma resposta lenta do sistema com efeito de cauda. A curva de resposta aproxima-se do sinal de referência de forma assintótica, porém de maneira muito lenta, caracterizando o chamado efeito cauda. Após um longo intervalo de tempo, a saída tende à referência, não havendo erro em regime permanente. Esse comportamento indica que o atraso de fase degrada o desempenho dinâmico do sistema, tornando a resposta mais lenta, embora não comprometa a eliminação do erro estacionário.

### 5.10.3 Implementação e Resultado Prático no Supervisório

Após ser obtido o desempenho esperado do controlador por simulação, foi realizada a aplicação do mesmo na plataforma de ensaios. Os parâmetros do controlador, ganho, polo, zero e período de amostragem, foram inseridos no supervisório, e em seguida foi aplicado um degrau de referência em 230 mm. A Figura 30 apresenta a resposta obtida experimento realizado.

Figura 30 – Tela do supervisório com parâmetros inseridos e resposta prática do sistema com o controlador de Atraso de Fase.



Fonte: Próprio autor.

A resposta obtida no supervisório durante o ensaio prático apresentou um comportamento coerente com a dinâmica lenta do processo e com as limitações impostas pelo controlador de atraso de fase. Foi definido um nível desejado de 230 mm, sendo o nível do tanque próximo de 200 mm no início da aplicação do controlador. A combinação entre a baixa velocidade de reação característica desse processo e a reduzida ação corretiva do compensador fez com que o nível continuasse diminuindo até aproximadamente 130 mm, antes que o controlador conseguisse inverter a tendência e iniciar a subida.

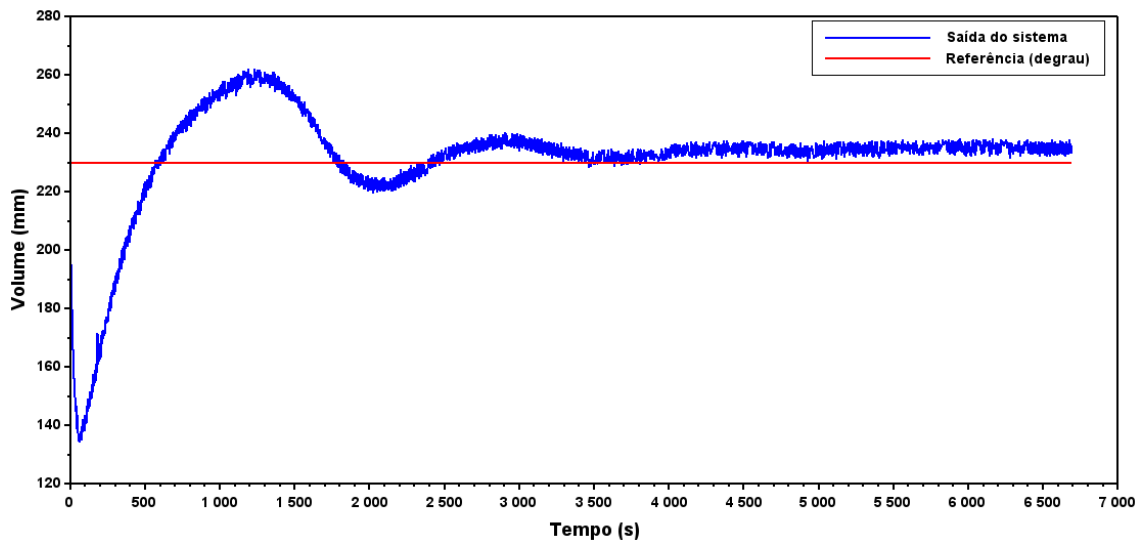
Após esse ponto, o sistema apresentou uma elevação relativamente lenta, seguida de um pico próximo de 260 mm, evidenciando um sobressinal consideravelmente alto, compatível com o observado em simulação. Em seguida, a resposta passou por um período de amortecimento gradual, aproximando-se da linha de referência de 230 mm, porém sem se estabilizar exatamente sobre ela. Assim como na simulação, a trajetória permanece sempre ligeiramente acima do valor desejado, evidenciando um erro estacionário positivo decorrente da redução do ganho em baixa frequência, típica do atraso de fase.

De forma geral, o comportamento prático confirma as tendências previstas no modelo, com um tempo de resposta elevado, sobressinal acentuado e sem atingir o estado estacionário de forma precisa, demonstrando que o controlador de atraso de fase não é o mais adequado em

comparação aos outros para a dinâmica deste sistema.

Os dados coletados pelo supervisor foram exportados em planilha e tratados no Scilab para permitir uma análise mais precisa. A Figura 31 mostra o gráfico reconstruído a partir desses dados.

Figura 31 – Resposta ao degrau obtida no Scilab a partir dos dados exportados do controlador de Atraso de Fase.

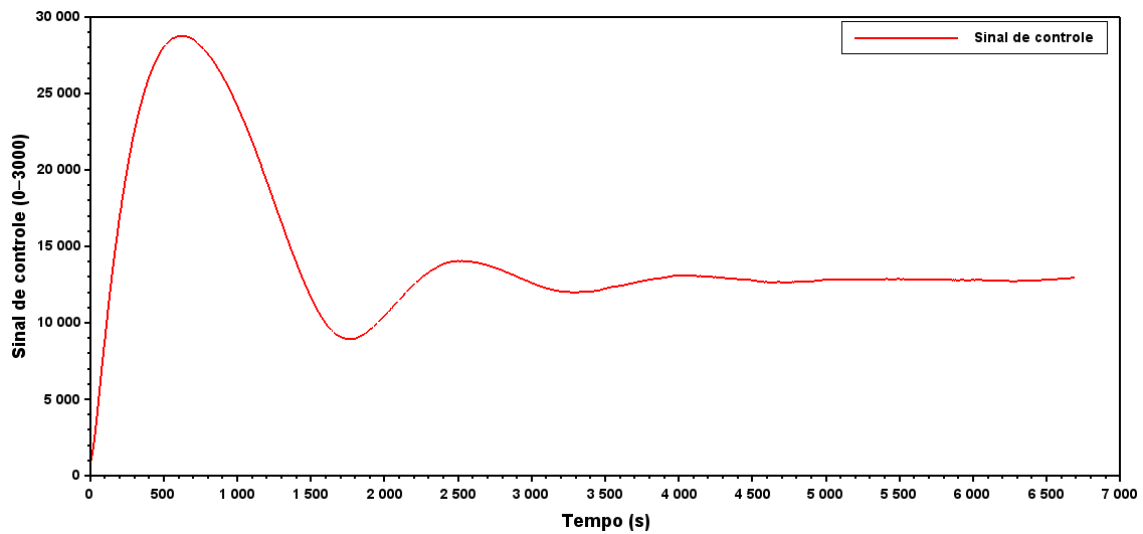


Fonte: Próprio autor.

A reconstrução do sinal no Scilab permite uma visualização mais clara da dinâmica do sistema, evidenciando de forma destacada o sobressinal elevado, a oscilação subsequente e o longo período de amortecimento até a aproximação da referência. Também se observa com precisão o erro estacionário positivo, que se mantém mesmo após o regime permanente. Esses resultados reforçam o comportamento identificado no ensaio prático e confirmam a coerência entre os dados experimentais e a resposta esperada para um controlador de atraso de fase aplicado a um processo de resposta lenta.

Também foi analisado o sinal de controle aplicado à bomba. A Figura 32 apresenta o sinal de controle ao longo da execução do algoritmo de compensação.

Figura 32 – Sinal de controle aplicado à bomba durante o processo de compensação - Controlador Atraso de Fase.



Fonte: Próprio autor.

O gráfico do sinal de controle da bomba, apresentado na Figura 31, mostra que ao longo do experimento, o sinal de controle aplicado iniciou em um valor relativamente baixo e aumentou de forma acentuada conforme o controlador buscava corrigir o grande erro inicial gerado pela queda do nível do tanque. Observa-se um pico expressivo, próximo ao valor máximo da bomba, seguido de oscilações amortecidas que refletem a dificuldade do compensador em modular a ação de controle em um processo extremamente lento. Após essa fase transitória, a atuação da bomba evolui para um regime com pequenas oscilações suaves.

Apesar da correta implementação do controlador de atraso de fase na plataforma, verificou-se que o seu desempenho prático ficou aquém do desejado. Devido à elevada vazão de saída do sistema, a potência da bomba precisa aumentar rapidamente para compensar a queda do nível, porém, o atraso de fase reduz o ganho do sistema na faixa de frequências associada à resposta em regime permanente. Como consequência, a reação do controlador ocorre de forma lenta, permitindo que o nível do tanque diminua antes que a bomba consiga elevar sua potência de maneira significativa. Além disso, o seguimento da referência apresentou baixa precisão, com longo tempo para aproximação do valor desejado e presença de erro estacionário positivo, caracterizado por uma aproximação lenta e assintótica ao valor de referência, fenômeno conhecido como efeito cauda, características esperadas quando se aplica atraso de fase em processos. Apesar dessas limitações de desempenho, a aplicação prática atingiu o objetivo do trabalho: demonstrar que a plataforma didática desenvolvida é capaz de implementar corretamente o controlador de atraso conforme o método apresentado, evidenciando, na prática, por que esse tipo de compensador não é adequado para esse processo.

## 6 CONCLUSÃO

O desenvolvimento da plataforma didática permitiu demonstrar, na prática, o processo de modelagem do sistema a partir da resposta em malha aberta, a realização dos ensaios de resposta ao degrau em malha fechada para os controladores de avanço de fase, atraso de fase e PID, bem como a avaliação de suas limitações diante da dinâmica do processo. A integração entre o código de discretização dos parâmetros, programado em Texto Estruturado no MasterTool, e a interface de operação via IHM mostrou-se eficiente, permitindo ao usuário inserir os parâmetros dos controladores no domínio contínuo, realizar a discretização dos controladores automaticamente e aplicar o controlador discretizado no sistema, além de enviar os dados dos ensaios para o supervisor, que também fornece a opção de exportação desses dados para análise externa à plataforma.

Os resultados experimentais confirmaram o comportamento previsto em simulação. O controlador de avanço apresentou melhor desempenho, com resposta mais rápida e menor erro. O controlador PI mostrou maior agressividade no sinal de controle, mas ainda assim conseguiu estabilizar o sistema. Já o controlador de atraso de fase apresentou desempenho limitado, com tempo de resposta elevado, erro estacionário e pouca capacidade de correção, comportamento esperado para processos com dinâmica lenta.

De modo geral, o projeto demonstrou que a plataforma é capaz de realizar a discretização e implementação correta de cada controlador, reproduzindo com precisão o comportamento do processo físico. Assim, o trabalho cumpre seu objetivo ao fornecer uma ferramenta prática para o estudo de controle clássico, permitindo ao usuário entender não só como projetar os controladores, mas também como as características do processo influenciam o desempenho obtido. Dessa forma, a plataforma mostra potencial para uso em atividades didáticas e laboratoriais, facilitando o aprendizado prático dos métodos de compensação e contribuindo para a formação na área de Controle e Automação.

## REFERÊNCIAS

- AGUIRRE, L. A. **Introdução à identificação de sistemas: técnicas lineares e não-lineares aplicadas a sistemas reais**. 3. ed. Belo Horizonte: Editora da UFMG, 2007. ISBN 9788570417874. Citado na página 24.
- AGUIRRE, L. A. **Controle de Sistemas Amostrados**. 1. ed. Rio de Janeiro: E-papers, 2020. ISBN 978-65-87065-05-2. Citado 3 vezes nas páginas 23, 28 e 29.
- ALTUS. **MasterTool IEC XE MT8500: Descrição do Produto**. Porto Alegre, 2021. Cod. Doc.: CT103705, Revisão M. Disponível em: <<https://www.altus.com.br>>. Citado na página 32.
- ALTUS S.A. **Manual de Utilização do BluePlant**. São Leopoldo, RS, 2018. Disponível em: <<https://www.altus.com.br>>. Acesso em: 9 nov. 2025. Citado na página 31.
- AMARAL, R. P.; ALMEIDA, G. M. de; SALLES, J. L. F.; MUNARO, C. J.; CUADROS, M. A. d. S. L. **Ambiente para Simulação e Aplicação Numa Planta Didática de um Controlador PID com Anti WIN-UP**. [S.l.]: Artigo apresentado no COBENGE–Instituto Federal do Espírito Santo. Blumenau, 2011. Citado na página 19.
- DORF, R. C.; BISHOP, R. H. **Sistemas de controle modernos**. 13. ed. Rio de Janeiro: LTC, 2020. ISBN 978-85-216-3512-3. Citado 4 vezes nas páginas 22, 24, 25 e 28.
- Exsto Tecnologia Ltda. **XC221: Banco de Ensaios para Controle de Nível e Vazão**. Santa Rita do Sapucaí, MG, 2021. Manual do usuário, Rev. B. Disponível em: <<http://www.exsto.com.br>>. Citado 3 vezes nas páginas 14, 15 e 33.
- FERREIRA, L. R. d. S. **Estudo e desenvolvimento de um sistema de controle discreto para interface homem-máquina touchscreen**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2016. Citado 2 vezes nas páginas 20 e 21.
- HOTZ, J. d. S. **Bancada didática para controle de nível e temperatura**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2014. Citado 2 vezes nas páginas 19 e 20.
- IDEALI, W. **Conectividade em Automação e IoT: Protocolos I2C, SPI, USB, TCP-IP entre outros**. 1. ed. Rio de Janeiro: Alta Books, 2021. ISBN 978-85-5081-345-5. Citado 2 vezes nas páginas 30 e 31.
- LUGLI, A. B.; SANTOS, M. M. D. **Redes Industriais para Automação Industrial: AS-I, PROFIBUS e PROFINET**. 2. ed. São Paulo: Editora Érica, 2023. ISBN 978-85-365-3202-8. Citado 2 vezes nas páginas 30 e 31.
- MAYA, P. A.; LEONARDI, F. **Controle Essencial**. São Paulo: Person Education do Brasil, 2014. ISBN 978-85-430-0241-5. Citado 3 vezes nas páginas 14, 22 e 31.
- MORAES, C. C. de; CASTRUCCI, P. de L. **Engenharia de Automação Industrial**. 2. ed. Rio de Janeiro: LTC, 2010. ISBN 978-85-216-1532-3. Citado 3 vezes nas páginas 14, 29 e 30.
- OGATA, K. **Engenharia de Controle Moderno**. São Paulo: Person Prentice Hall, 2010. ISBN 8576058103. Citado 8 vezes nas páginas 14, 22, 23, 24, 25, 26, 27 e 56.
- PIMENTEL, J. F. Desenvolvimento de um protótipo de controle de tanques acoplados para ensino de engenharia. Universidade Federal de Uberlândia, 2024. Citado na página 19.

PLCopen. **IEC 61131-3: A norma para programação**. Eindhoven, 2004. Tradução: Marcos Fonseca. Disponível em: <<https://www.plcopen.org>>. Citado na página 32.

RIBEIRO, R. E. M.; ABREU, C. R. S. de. **Inovação em Sistemas de Produção na Era da Indústria 4.0**. 1. ed. Teresina, PI: KDP, 2020. ISBN 9798651411368. Citado 3 vezes nas páginas 14, 19 e 30.

Rücken Automação. **Transmissor de Pressão Modelo RTP-420**. [S.l.], 2020. Datasheet técnico. Disponível em: <<https://www.rucken.com.br>>. Citado na página 33.

SAUER, C.; PAIXÃO, E. P.; SILVA, J. V. das C.; REIMBOLD, M. M. Plataforma didática para o ensino de controle. 2017. Citado na página 19.

SILVA, G. G. de O.; CAMUSSO, D.; BRUNO, D. O. T.; FERRER, J. A. G. Indústria 4.0: Um estudo sobre sua aplicação no ensino técnico profissionalizante. **Revista bras. de Mecatrônica**, p. 1–14, 2019. Citado na página 16.

## APÊNDICE A – CÓDIGO-FONTE DO SISTEMA EM TEXTO ESTRUTURADO (ST)

Este apêndice apresenta o código desenvolvido em linguagem Texto Estruturado (ST) no *software* MasterTool, utilizado para a implementação da lógica de controle e aquisição de dados do sistema. São apresentadas as declarações de variáveis globais e locais, seguidas do trecho principal do código.

### A.1 Declaração de Variáveis Globais

```

1 VAR_GLOBAL
2 //BOTOES DE ATIVACAO DOS COMANDOS
3     ATIVA_PID                : BOOL;
4     DESATIVA_PID            : BOOL;
5     ATIVA_AV                : BOOL;
6     DESATIVA_AV            : BOOL;
7     ATIVA_AT                : BOOL; // ATRASO DE FASE
8     DESATIVA_AT            : BOOL;
9     ATIVA_MA                : BOOL;
10    DESATIVA_MA            : BOOL;
11
12    //DADOS MALHA ABERTA
13    Nivel_MA                : REAL;
14    Tempo_MA                : REAL;
15    Pot_bomba                : INT;
16    debral_MA                : REAL;
17
18    //DADOS PARA O CONTROLADOR PID
19    Ts                        : INT; // Periodo de
        amostragem (em segundos);
20    Kp                        : REAL; // Ganho
        Proporcional
21    Ti                        : REAL;
22    Ki                        : REAL; // (KP/Ti)
23    Nivel_PID                : REAL;
24    Tempo_PID                : REAL;
25    Nivel_Desej_PID          : REAL;
26    Bomba_PID                : INT;
27
28    // DADOS AVANCO
29    Ts_AV                    : INT;
30    Polo                      : REAL; //POLO

```

```

31     Zero                                : REAL; //ZERO
32     K                                    : REAL; // GANHO
33     Nivel_AV                             : REAL;
34     Tempo_AV                             : REAL;
35     Nivel_Desej_AV                       : REAL;
36     Bomba_AV                             : INT;
37
38     // DADOS ATRASO
39     Ts_AT                                : INT;
40     Polo_AT                              : REAL; //POLO
41     Zero_AT                              : REAL; //ZERO
42     K_AT                                  : REAL; // GANHO
43     Nivel_AT                             : REAL;
44     Tempo_AT                             : REAL;
45     Nivel_Desej_AT                       : REAL;
46     Bomba_AT                             : int;
47
48 END_VAR

```

## A.2 Declaração de Variáveis Locais

```

1 VAR
2     Nivel_Atual                          : REAL;
3     D                                      : REAL;
4     // variaveis da funcao de calibracao
5     a                                      : REAL := 0.0175797;
6     b                                      : REAL := 22.3560130;
7     // variaveis do calculo do erro
8     ek                                    : REAL := 0.0; // erro atual
9     ek_1                                  : REAL := 0.0; // erro anterior
10    ek_2                                   : REAL := 0.0;
11    // Variavel do sinal de controle
12    uk                                      : REAL := 0.0; // sinal de controle
13    uk_1                                    : REAL := 0.0; // sinal de controle
14    anterior
15    uk_2                                    : REAL := 0.0;
16    uk_Int                                  : INT; // sinal de controle inteiro
17
18    Primeira_exec                          : BOOL := TRUE;
19    TON_MA                                  : TON;
20    TON_AV                                  : TON;

```

```
20     TON_AT           : TON;
21     TON_PID         : TON;
22     Tempo           : int;
23     x               : TIME;
24
25     TEMPO_ENVIO_AV  : TON;
26     TEMPO_EV        : TIME;
27     // Dados para calcular controlador de Avanco
28     A1              : REAL; //
29     A2              : REAL;
30     B0              : REAL;
31     B1              : REAL;
32     B2              : REAL;
33
34     SEL01           : BOOL;
35     SEL02           : BOOL;
36     SEL03           : BOOL;
37
38     temp            : TIME;
39     SEL04           : BOOL;
40     PWM_Q14        : LibIntegratedIo.PWM;
41     Cal_PWM        : REAL;
42     uk_PWM         : REAL;
43     Duty           : USINT;
44 END_VAR
```

### A.3 Trecho principal do código

```
1 //-----
2 // RESPOSTA AO DEGRAU EM MALHA ABERTA
3 SEL01 := (SEL01 OR VG.ATIVA_MA) AND NOT (VG.DESATIVA_MA OR SEL02 OR
4     SEL03 OR SEL04);
5
6 IF SEL01 OR i00 THEN
7
8     IF Primeira_exec THEN
9         Primeira_exec:= FALSE;
10        Nivel_Atual := a* AI0 + b; // calibacao da leitura
11
12        VG.Nivel_MA := Nivel_Atual;
13        VG.Tempo_MA := Tempo;
```

```
12
13         // AJUSTA VALOR DA BOMBA PARA QUE NAO SEJA MAIOR DE
           30000 E MENOR QUE 0
14     IF VG.Pot_bomba > 30000 THEN
15     VG.Pot_bomba := 30000;
16
17     ELSIF VG.Pot_bomba < 0 THEN
18     VG.Pot_bomba := 0;
19     END_IF
20
21     Cal_PWM := (TO_REAL(VG.Pot_bomba) / 30000.0) * 100.0;
22     Duty:= REAL_TO_USINT(Cal_PWM);
23
24     PWM_Q14(ENABLE := TRUE,
25             FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
26             FREQUENCY := 1000,
27             DUTY_CYCLE := Duty);
28 END_IF
29
30     temp:= INT_TO_TIME((VG.Ts*1000)); // CONVERTE O VALOR DO
           PERIODO ESTABELECIDO PARA TIME
31     TON_MA(IN:=TRUE , PT := temp); // Temporiza o periodo
           escolhido
32     x:=TON_MA.ET;
33
34     IF TON_MA.Q THEN
35         Nivel_Atual := a* AI0 + b; // calibacao da leitura
36         VG.Nivel_MA := Nivel_Atual;
37         Tempo := (Tempo + VG.Ts);
38         VG.Tempo_MA := Tempo;
39
40         IF VG.Pot_bomba > 30000 THEN
41         VG.Pot_bomba := 30000;
42         ELSIF VG.Pot_bomba < 0 THEN
43         VG.Pot_bomba := 0;
44         END_IF
45
46         Cal_PWM := (TO_REAL(VG.Pot_bomba) / 30000.0) * 100.0;
47         Duty:= REAL_TO_USINT(Cal_PWM);
48         // manda o valor pra saida pwm
49         PWM_Q14(ENABLE := TRUE,
50                 FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
```

```
51             FREQUENCY := 1000,
52             DUTY_CYCLE := Duty);
53
54             TON_MA (IN:=FALSE);
55             TON_MA (IN:=TRUE);
56         END_IF
57
58 END_IF
59
60 //-----
61 // CONTROLADOR DE AVANCO DE FASE
62 SELO2 := (SELO2 OR VG.ATIVA_AV OR I05) AND NOT VG.DESATIVA_AV;
63 IF SELO2 THEN
64
65     D := 2*VG.Ts_AV*VG.Polo + 4;    // calcule com T, p
66     A1 := 8.0 / D;
67     A2 := (2*VG.Ts_AV*VG.Polo - 4.0) / D;
68     B0 := VG.K*VG.Ts_AV*(VG.Ts_AV*VG.Zero + 2.0) / D;
69     B1 := 2.0*VG.K*VG.Ts_AV*VG.Ts_AV*VG.Zero / D;
70     B2 := VG.K*VG.Ts_AV*(VG.Ts_AV*VG.Zero - 2.0) / D;
71
72     // FAZ A PRIMEIRA LEITURA SEM USAR O TEMPORIZADOR
73     IF Primeira_exec THEN
74         Primeira_exec:= FALSE;
75         Nivel_Atual := a* AI0 + b; // calibacao da leitura
76         ek := VG.Nivel_Desej_AV - Nivel_Atual; // Calculo do
           erro
77
78         uk := A1*uk_1 + A2*uk_2 + B0*ek + B1*ek_1 + B2*ek_2;
79
80         ek_1:=ek;
81         uk_1:=uk;
82
83
84         IF uk > 30000 THEN
85             uk := 30000;
86         ELSIF uk < 0 THEN
87             uk := 0;
88         END_IF
89
```

```
90         VG.Bomba_AV:=REAL_TO_INT(uk); // envia potencia da
          bomba pro supervisorio
91
92         uk_PWM := ((uk) / 30000.0) * 100.0;
93         Duty:= REAL_TO_USINT(uk_PWM);
94
95         PWM_Q14(ENABLE := TRUE,
96                 FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
97                 FREQUENCY := 1000,
98                 DUTY_CYCLE := Duty);
99
100        // manda dados pro supervisorio
101        VG.Nivel_AV := Nivel_Atual;
102        VG.Tempo_AV := Tempo;
103    END_IF
104
105    temp:= INT_TO_TIME((VG.Ts_AV *1000)); // CONVERTE O VALOR DO
          PERIODO ESTABELECIDO PARA TIME
106    TON_AV(IN:=TRUE , PT := temp); // Temporiza o intervalo de
          calculo
107    x:=TON_AV.ET;
108
109    IF TON_AV.Q THEN
110        Nivel_Atual := a* AI0 + b; // calibacao da leitura
111        ek := VG.Nivel_Desej_AV - Nivel_Atual; // Calculo do
          erro
112
113        uk := A1*uk_1 + A2*uk_2 + B0*ek + B1*ek_1 + B2*ek_2;
114
115        uk_2 := uk_1;
116        uk_1 := uk;
117        ek_2 := ek_1;
118        ek_1 := ek;
119
120        IF uk > 30000 THEN
121            uk := 30000;
122        ELSIF uk < 0 THEN
123            uk := 0;
124        END_IF
125
126        VG.Bomba_AV:=REAL_TO_INT(uk); // envia potencia da
          bomba pro supervisorio
```

```
127
128         uk_PWM := ((uk) / 30000.0) * 100.0;
129         Duty:= REAL_TO_USINT(uk_PWM);
130
131         PWM_Q14(ENABLE := TRUE,
132               FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
133               FREQUENCY := 1000,
134               DUTY_CYCLE := Duty);
135
136         // MANDA VALORES DE TEMPO E NIVEL PARA O SUPERVISORIO
137         VG.Nivel_AV := Nivel_Atual;
138         Tempo := (Tempo + VG.Ts_AV);
139         VG.Tempo_AV := Tempo;
140
141         TON_AV(IN:=FALSE);
142         TON_AV(IN:=TRUE);
143     END_IF
144
145 END_IF
146
147 //-----
148 // CONTROLADOR DE ATRASO DE FASE
149
150 SELO3 := (SELO3 OR VG.ATIVA_AT) AND NOT VG.DESATIVA_AT;
151 IF SELO3 THEN
152
153     D := 2*VG.Ts_AT*VG.Polo_AT + 4;    // calcule com T, p
154     A1 := 8.0 / D;
155     A2 := (2*VG.Ts_AT*VG.Polo_AT - 4.0) / D;
156     B0 := VG.K_AT*VG.Ts_AT*(VG.Ts_AT*VG.Zero_AT + 2.0) / D;
157     B1 := 2.0*VG.K_AT*VG.Ts_AT*VG.Ts_AT*VG.Zero_AT / D;
158     B2 := VG.K_AT*VG.Ts_AT*(VG.Ts_AT*VG.Zero_AT - 2.0) / D;
159
160     // FAZ A PRIMEIRA LEITURA SEM USAR O TEMPORIZADOR
161     IF Primeira_exec THEN
162         Primeira_exec:= FALSE;
163         Nivel_Atual := a* AI0 + b; // calibacao da leitura
164         ek := VG.Nivel_Desej_AT - Nivel_Atual; // Calculo do
           erro
165
166         uk := A1*uk_1 + A2*uk_2 + B0*ek + B1*ek_1 + B2*ek_2;
```

```
167
168         ek_1:=ek;
169         uk_1:=uk;
170
171         IF uk > 30000 THEN
172             uk := 30000;
173         ELSIF uk < 0 THEN
174             uk := 0;
175         END_IF
176
177         VG.Bomba_AT:=REAL_TO_INT(uk); // envia potencia da
178             bomba pro supervisorio
179
180         uk_PWM := ((uk) / 30000.0) * 100.0;
181         Duty:= REAL_TO_USINT(uk_PWM);
182
183         PWM_Q14(ENABLE := TRUE,
184             FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
185             FREQUENCY := 1000,
186             DUTY_CYCLE := Duty);
187
188         // manda dados pro supervisorio
189         VG.Nivel_AT := Nivel_Atual;
190         VG.Tempo_AT := Tempo;
191     END_IF
192
193     temp:= INT_TO_TIME((VG.Ts_AT *1000)); // CONVERTE O VALOR DO
194         PERIODO ESTABELECIDO PARA TIME
195     TON_AT(IN:=TRUE , PT := temp); // Temporiza o periodo
196         escolhido
197     x:=TON_AT.ET;
198
199     IF TON_AT.Q THEN
200         Nivel_Atual := a* AI0 + b; // calibacao da leitura
201         ek := VG.Nivel_Desej_AT - Nivel_Atual; // Calculo do
202             erro
203
204         uk := A1*uk_1 + A2*uk_2 + B0*ek + B1*ek_1 + B2*ek_2;
205
206         uk_2 := uk_1;
207         uk_1 := uk;
208         ek_2 := ek_1;
```

```
205         ek_1 := ek;
206
207         IF uk > 30000 THEN
208             uk := 30000;
209         ELSIF uk < 0 THEN
210             uk := 0;
211         END_IF
212
213         VG.Bomba_AT:=REAL_TO_INT(uk); // envia potencia da
                bomba pro supervisorio
214
215         uk_PWM := ((uk) / 30000.0) * 100.0;
216         Duty:= REAL_TO_USINT(uk_PWM);
217
218         PWM_Q14(ENABLE := TRUE,
219                FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
220                FREQUENCY := 1000,
221                DUTY_CYCLE := Duty);
222
223         // MANDA VALORES DE TEMPO E NIVEL PARA O SUPERVISORIO
224         VG.Nivel_AT := Nivel_Atual;
225         Tempo := (Tempo + VG.Ts_AT);
226         VG.Tempo_AT := Tempo;
227
228         TON_AT(IN:=FALSE);
229         TON_AT(IN:=TRUE);
230     END_IF
231     //TON_AV(IN:=TRUE);
232
233 END_IF
234
235
236 //-----
237 // CONTROLADOR PID
238 SELO4 := (SELO4 OR VG.ATIVA_PID) AND NOT VG.DESATIVA_PID; // SELO DE
                ATIVACAO QUANDO PARA MANTER ATIVADO QUANDO O BOTAO FOR SOLTO
239 IF SELO4 OR I01 THEN // ACIONAMENTO DO CONTROLADOR PID
240
241     VG.Ki :=(VG.Kp / VG.Ti); //CALCULA O VALOR DE KI
242
243     // FAZ A PRIMEIRA LEITURA SEM USAR O TEMPORIZADOR
```

```
244     IF Primeira_exec THEN
245         Primeira_exec:= FALSE;
246         Nivel_Atual := a* AI0 + b; // calibraçAO da leitura
247         ek := VG.Nivel_Desej_PID - Nivel_Atual; // Calculo do
           erro
248
249         uk:= uk_1 + (VG.Kp + VG.Ki * VG.Ts / 2)*ek + (-VG.Kp
           + VG.Ki * VG.Ts/2)*ek_1;
250
251         ek_1:=ek;
252         uk_1:=uk;
253
254         IF uk > 30000 THEN
255             uk := 30000;
256         ELSIF uk < 0 THEN
257             uk := 0;
258         END_IF
259
260         VG.Bomba_PID:=REAL_TO_INT(uk); // envia potencia da
           bomba pro supervisorio
261
262         uk_PWM := ((uk) / 30000.0) * 100.0;
263         Duty:= REAL_TO_USINT(uk_PWM);
264
265         PWM_Q14(ENABLE := TRUE,
266                 FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
267                 FREQUENCY := 1000,
268                 DUTY_CYCLE := Duty);
269
270         // manda dados pro supervisorio
271         VG.Nivel_PID := Nivel_Atual;
272         VG.Tempo_PID := Tempo;
273     END_IF
274
275     temp:= INT_TO_TIME((VG.Ts*1000)); // CONVERTE O VALOR DO
           PERIODO ESTABELECIDO PARA TIME
276     TON_PID(IN:=TRUE , PT := temp); // Temporiza o periodo
           escolhido
277     x:=TON_PID.ET;
278
279     IF TON_PID.Q THEN
280         Nivel_Atual := a* AI0 + b; // calibração da leitura
```

```
281
282         ek := VG.Nivel_Desej_PID - Nivel_Atual; // Calculo do
           erro
283
284         // funcao de discretizacao do controlador PID
285     //     uk := uk_1 + ( Kp*(ek - ek_1)) + (Ki*Ts*ek) + ((Kd/Ts
           )*(ek-2*ek_1 + ek_2));
286         uk:= uk_1 + (VG.Kp + VG.Ki * VG.Ts / 2)*ek + (-VG.Kp
           + VG.Ki * VG.Ts/2)*ek_1;
287
288         IF uk > 30000 THEN
289             uk := 30000;
290         ELSIF uk < 0 THEN
291             uk := 0;
292         END_IF
293
294         VG.Bomba_PID:=real_to_int(uk); // envia potencia da
           bomba pro supervisorio
295
296         uk_PWM := ((uk) / 30000.0) * 100.0;
297         Duty:= REAL_TO_USINT(uk_PWM);
298
299         PWM_Q14(ENABLE := TRUE,
300                FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
301                FREQUENCY := 1000,
302                DUTY_CYCLE := Duty);
303
304         // MANDA VALORES DE TEMPO E NIVEL PARA O SUPERVISORIO
305         VG.Nivel_PID := Nivel_Atual;
306         Tempo := (Tempo + VG.Ts);
307         VG.Tempo_PID := Tempo;
308         ek_1:=ek; // atualiza o erro anterior
309         uk_1:=uk;
310
311         TON_PID(IN:=FALSE);
312         TON_PID(IN:=TRUE);
313     END_IF
314     //TON_PID(IN:=TRUE);
315
316 END_IF
317
```

```
318 // TODA VEZ QUE DESATIVAR UM COMANDO RESETA VARIÁVEIS TEMPO E REATIVA
    PRIMEIRA LEITURA
319 IF VG.DESATIVA_MA OR VG.DESATIVA_PID OR VG.DESATIVA_AT OR VG.
    DESATIVA_AV THEN
320     Tempo:=0;
321     Primeira_exec:= TRUE;
322     uk_1:=0;
323     uk_2:=0;
324     ek_1:=0;
325     ek_2:=0;
326
327
328     PWM_Q14(ENABLE := TRUE,
329     FAST_OUTPUT := IntegratedIO.FastOutputs.Q14,
330     FREQUENCY := 1000,
331     DUTY_CYCLE := 0);
332 END_IF
333 //-----
```