

Ulisses Maynarth Souza Soares Viana

**Aplicação de Controle Clássico com Arduino, Simulink e Matlab  
para Auxílio no Ensino**

Brasil

2024

Ulisses Maynarth Souza Soares Viana

**Aplicação de Controle Clássico com Arduino, Simulink e Matlab  
para Auxílio no Ensino**

Trabalho de Conclusão de Curso apresentado à banca avaliadora do curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Amazonas como pré-requisito para obtenção do título de Engenheiro de Controle e Automação.

Universidade do Estado do Amazonas – UEA

Escola Superior de Tecnologia – EST

Engenharia de Controle e Automação

Orientador: Prof. Dr. José Ruben Sicchar Vilchez

Brasil

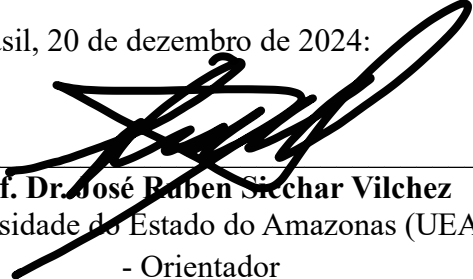
2024

Ulisses Maynarth Souza Soares Viana

## **Aplicação de Controle Clássico com Arduino, Simulink e Matlab para Auxílio no Ensino**


Trabalho de Conclusão de Curso apresentado à banca avaliadora do curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Amazonas como pré-requisito para obtenção do título de Engenheiro de Controle e Automação.

Trabalho Aprovado. Brasil, 20 de dezembro de 2024:




---

**Prof. Dr. José Ruben Sicchar Vilchez**  
Universidade do Estado do Amazonas (UEA)  
- Orientador



---

**Prof. Dr. Rodrigo Farias Araújo**  
Universidade do Estado do Amazonas (UEA)



---

**Prof. Dr. Florindo Antonio de Carvalho  
Ayres Júnior**  
Universidade Federal do Amazonas (UFAM)



---

**Prof. Dr. Israel Mazaira Morales**  
Universidade do Estado do Amazonas (UEA)

Brasil

2024

## **Agradecimentos**

Agradeço primeiramente a Deus, por me conceder saúde, força e perseverança para alcançar mais esta etapa significativa da minha vida. Sem sua graça, sabedoria, proteção e soberana vontade, nada disso seria possível.

Agradeço à minha família, pelo apoio e cuidado no decorrer da minha vida, e por terem me dado condições de chegar até aqui.

Ao meu orientador, Prof. Dr. José Ruben Sicchar, grato pela sua incansável dedicação e trabalho árduo em prol do curso de Engenharia de Controle e Automação. Sua orientação e contribuição durante minha trajetória acadêmica foram fundamentais para o meu crescimento pessoal e profissional.

À Universidade do Estado do Amazonas, por ter me possibilitado muitas oportunidades de aprendizado durante a realização deste curso.

Aos amigos e professores do curso de Engenharia de Controle e Automação, minha gratidão pela constante troca de conhecimentos, pelo apoio mútuo e pelos momentos de aprendizado compartilhados ao longo dessa jornada.

Por fim, expresso minha gratidão a todas as pessoas que, de alguma forma, estiveram ao meu lado durante essa caminhada.

*“aquele que tem, ele só, a imortalidade e habita na luz inacessível; a quem nenhum dos homens viu nem pode ver; ao qual seja honra e poder sempiterno. Amém!”*

*(1Tm 6.16)*

## Resumo

O presente trabalho apresenta uma aplicação prática de controle clássico utilizando Arduino, Simulink e MATLAB, voltada para o ensino de estudantes de Engenharia de Controle e Automação. O sistema experimental é composto por um motor DC com encoder, Arduino Uno, fonte de bancada Yaxun PS-1502DD+ e ponte H L298N como planta de controle. A proposta conecta teoria e prática ao permitir que os alunos desenvolvam modelos matemáticos ou identifiquem experimentalmente a planta para obter a função de transferência do sistema. Com o suporte da comunicação serial entre o Simulink e o Arduino, é possível implementar o controle e analisar a resposta dinâmica em tempo real, comparando diretamente o comportamento modelado com o desempenho real do sistema. Essa abordagem permite que os estudantes realizem ajustes no controle e analisem o impacto dessas alterações no desempenho, além de facilitar adaptações para diferentes tipos de sistemas, dado que uma base prática já foi estabelecida. O método visa tornar o aprendizado mais dinâmico e interativo, despertando maior interesse pela área e promovendo a aplicação de conceitos teóricos em situações reais. Além disso, busca superar deficiências identificadas em cursos que, frequentemente, restringem-se aos cálculos teóricos sem apresentar implementações práticas, fomentando, assim, o desenvolvimento de competências fundamentais para a formação de futuros engenheiros.

**Palavras-chave:** Controle Clássico, Arduino, Ensino, Sistemas Dinâmicos.

## **Abstract**

The present work presents a practical application of classical control using Arduino, Simulink and MATLAB, aimed at teaching Control and Automation Engineering students. The experimental system consists of a DC motor with encoder, Arduino Uno, Yaxun PS-1502DD+ bench source and L298N H bridge as control plant. The proposal connects theory and practice by allowing students to develop mathematical models or experimentally identify the plant to obtain the system's transfer function. With the support of serial communication between Simulink and Arduino, it is possible to implement control and analyze the dynamic response in real time, directly comparing the modeled behavior with the real system performance. This approach allows students to make control adjustments and analyze the impact of these changes on performance, in addition to facilitating adaptations to different types of systems, given that a practical basis has already been established. The method aims to make learning more dynamic and interactive, awakening greater interest in the area and promoting the application of theoretical concepts in real situations. Furthermore, it seeks to overcome deficiencies identified in courses that are often restricted to theoretical calculations without presenting practical implementations, thus promoting the development of fundamental skills for the training of future engineers.

**Keywords:** Classical Control, Arduino, Teaching, Dynamic Systems.

## Lista de Ilustrações

Figura 1 – Diagrama Malha Aberta .....	18
Figura 2 – Diagrama Malha Fechada .....	19
Figura 3 – Medidas de Desempenho para um Sistema Subamortecido .....	22
Figura 4 – Curva de formato S a resposta ao degrau unitário de uma planta.....	24
Figura 5 – Diagrama de Blocos Controlador PID Paralelo .....	29
Figura 6 – Diagrama de Blocos Controlador PID Série .....	29
Figura 7 – Diagrama de Blocos Controlador PID de uma Planta Genérica .....	29
Figura 8 – Exemplo de um Sistema Elétrico e Mecânico de um Motor CC .....	33
Figura 9 – Exemplo de um Diagrama de Blocos de um Motor CC.....	34
Figura 10 – Arduino Uno.....	36
Figura 11 – IDE Arduino .....	39
Figura 12 – Bibliotecas Simulink .....	41
Figura 13 – Fluxograma de funcionamento do Simulink Support Package for Arduino .....	44
Figura 14 – Fluxograma Geral.....	45
Figura 15 – Motor DC 6V 280 RPM com Roda e Encoder.....	46
Figura 16 – Motor DC 6V 280 RPM Dimensões .....	47
Figura 17 – Motor DC 6V 280 RPM Especificações .....	47
Figura 18 – Ponte H.....	48
Figura 19 – Fonte de Bancada .....	48
Figura 20 – Diagrama de Montagem do Hardware do Sistema de Controle.....	49
Figura 21 – Hardware do Sistema Montado.....	50
Figura 22 – Diagrama esquemático de um motor CC .....	50
Figura 23 – Diagrama Iterativo no Simulink em Malha Aberta .....	55
Figura 24 – Curva de Reação .....	56
Figura 25 – Funções de Transferência com Atraso.....	62
Figura 26 – Obtendo frequência de corte do sistema .....	63
Figura 27 – Diagrama em Malha Fechada.....	65
Figura 28 – Subsistema / Planta Real .....	66
Figura 29 – Subsistema / Modelo Matemático .....	67
Figura 30 – Subsistema / Modelo Experimental .....	67
Figura 31 – Ajustes Regressivos.....	69

Figura 32 – Resposta em Malha Aberta sem Regressão.....	70
Figura 33 – Resposta em Malha Aberta com Regressão Linear.....	70
Figura 34 – Resposta em Malha Aberta com Regressão Polinomial de 3ª Ordem.....	71
Figura 35 – Resposta em Malha Aberta com Regressão Polinomial de 3ª Ordem com Ganho Ajustado.....	71
Figura 36 – Resposta em Malha Fechada - P - Degrau (16.59 rad/s); $K_p = 1.107$ .....	73
Figura 37 – Resposta em Malha Fechada - PI - Degrau (16.59 rad/s) .....	73
Figura 38 – Resposta em Malha Fechada - PID - Degrau (16.59 rad/s).....	74
Figura 39 – Resposta em Malha Fechada - P - Rampa (0 a 27,25 rad/s).....	76
Figura 40 – Resposta em Malha Fechada - PI - Rampa (0 a 27,25 rad/s).....	76
Figura 41 – Resposta em Malha Fechada - PID - Rampa (0 a 27,25 rad/s) .....	77
Figura 42 – Resposta em Malha Fechada - P - Entrada Senoidal (Velocidade) .....	78
Figura 43 – Resposta em Malha Fechada - PI - Entrada Senoidal (Velocidade).....	79
Figura 44 – Resposta em Malha Fechada - PID - Entrada Senoidal (Velocidade).....	79
Figura 45 – Resposta em Malha Fechada - Controlador Ressonante .....	81
Figura 46 – Resposta em Malha Fechada PID - Sem Ajustes .....	83
Figura 47 – Resposta em Malha Fechada PID - Com Ajustes.....	83
Figura 48 – Resposta em Malha Fechada PI - Sem Ajustes .....	84
Figura 49 – Resposta em Malha Fechada PI - COM Ajustes.....	85
Figura 50 – Lugar das Raízes .....	86
Figura 51 – Diagrama de Bode.....	87
Figura 52 – Conceitos de Controle na Prática .....	89

## **Lista de Tabelas**

Tabela 1 – Regra de sintonia de Ziegler- Nichols para o método da curva de reação.....	26
Tabela 2 – Especificações Técnicas do Arduino.....	37
Tabela 3 – Lista completa dos parâmetros.....	57
Tabela 4 – Resumo de dados para Regressão .....	68

## Lista de Abreviaturas e Siglas

PWM	<i>Pulse Width Modulation</i>
IDE	<i>Integrated Development Environment</i>
PID	<i>Proportional-Integral-Derivative</i>
PI	<i>Proportional-Integral</i>
P	<i>Proportional</i>
DC	<i>Direct Current</i>
IDE	<i>Integrated Development Environment</i>
ADC	<i>Analog-to-Digital Converter</i>
DAC	<i>Digital-to-Analog Converter</i>
HMI	<i>Human-Machine Interface</i>
MATLAB	<i>Matrix Laboratory</i>
IoT	<i>Internet of Things</i>
USB	<i>Universal Serial Bus</i>
UART	<i>Universal Asynchronous Receiver Transmitter</i>
I2C	<i>Inter-Integrated Circuit</i>
SPI	<i>Serial Peripheral Interface</i>
LED	<i>Light Emitting Diode</i>
MCU	<i>Microcontroller Unit</i>

# Sumário

<b>1</b>	<b>Introdução</b>	<b>13</b>
<b>1.1</b>	<b>Objetivos</b>	<b>14</b>
1.1.1	Objetivo Geral	14
1.1.2	Objetivos Específicos	15
<b>1.2</b>	<b>Estrutura do Trabalho</b>	<b>15</b>
<b>2</b>	<b>Referencial Teórico</b>	<b>16</b>
<b>2.1</b>	<b>Introdução ao Controle</b>	<b>16</b>
2.1.1	Função de Transferência	16
2.1.2	Malha Aberta	17
2.1.3	Malha Fechada	18
2.1.4	Estabilidade	20
2.1.5	Parâmetros de Desempenho de Sistemas de 2ª Ordem	20
2.1.6	Controladores PID	23
2.1.6.1	Método da Curva de Reação	24
2.1.6.2	Controlador Proporcional	26
2.1.6.3	Controlador Proporcional Integral	27
2.1.6.4	Controlador Proporcional Integral Derivativo	27
<b>2.2</b>	<b>Ensino de Controle Clássico</b>	<b>29</b>
2.2.1	Importância do Ensino Prático em Engenharia de Controle	30
2.2.2	Desafios no Ensino de Controle Clássico	31
2.2.3	Ferramentas Tradicionais de Ensino de Controle	32
<b>2.3</b>	<b>Arduino</b>	<b>34</b>
2.3.1	Definição e Principais Conceitos	35
2.3.2	Histórico	35
2.3.3	Vantagens	35
2.3.4	<i>Hardware</i>	36
2.3.5	<i>Software</i>	38
2.3.5.1	IDE	38
<b>2.4</b>	<b>Matlab e Simulink</b>	<b>40</b>
2.4.1	Integração do Matlab com Simulink	41
2.4.2	Aplicações e Recursos do Simulink	41
2.4.3	Simulink Coder	42

2.4.3.1 Principais Recursos do Simulink Coder .....	42
2.4.3.2 Aplicações e Benefícios.....	43
2.4.4 Pacotes de Suporte para Arduino .....	43
2.4.4.1 Pacotes de Suporte do Simulink e Geração de Código .....	44
2.4.4.2 Pacotes Adicionais e seus Benefícios .....	44
<b>3 Materiais e Métodos .....</b>	<b>45</b>
<b>3.1 Descrição do Sistema Experimental.....</b>	<b>45</b>
3.1.1 Componentes Utilizados .....	46
3.1.1.1 Motor JGA25-370 DC6V280RPM.....	46
3.1.1.2 Ponte H L298N.....	47
3.1.1.3 Fonte de Bancada Yanxun OS-1502DD+ .....	48
3.1.2 Montagem.....	49
<b>3.2 Desenvolvimento do Modelo Matemático.....</b>	<b>50</b>
3.2.1 Função de Transferência do Motor.....	52
3.2.2 Parâmetros do motor estabelecidos através de ensaios .....	53
<b>3.3 Comunicação Serial: Arduino e Simulink.....</b>	<b>55</b>
<b>3.4 Definindo Função de Transferência Experimentalmente .....</b>	<b>58</b>
<b>3.5 Parâmetros dos Controladores.....</b>	<b>59</b>
<b>3.6 Ajustes e Complementos no Sistema.....</b>	<b>61</b>
3.6.1 Funções de Transferência com atraso através da aproximação de pade.....	61
3.6.2 Filtro passa-baixa para suavização do sinal.....	62
3.6.3 Diagrama em Malha Fechada no Simulink e Gráficos via Matlab .....	64
3.6.4 Relação não linear entre o sinal PWM e a tensão enviada para o motor.....	67
<b>4 Resultados .....</b>	<b>72</b>
<b>4.1 Análise da Resposta Dinâmica do Sistema Comparando os Modelos .....</b>	<b>72</b>
4.1.1 Entrada Degrau.....	72
4.1.2 Entrada Rampa .....	75
4.1.3 Entrada Senoidal.....	78
<b>4.2 Impacto dos Ajustes no Desempenho do Sistema .....</b>	<b>82</b>
<b>4.3 Lugar das Raízes e Diagrama de Bode .....</b>	<b>85</b>
<b>4.4 Impacto Educacional e Aprendizagem dos Alunos.....</b>	<b>88</b>
<b>5 Conclusões.....</b>	<b>91</b>
<b>5.1 Trabalhos Futuros .....</b>	<b>91</b>
<b>Referências .....</b>	<b>93</b>

# 1 Introdução

A evolução tecnológica e a demanda por profissionais qualificados destacam a importância do ensino prático em Engenharia. No entanto, muitas instituições ainda focam em teoria e simulações, sem oferecer experiências práticas, o que prejudica a assimilação dos conteúdos e pode reduzir o interesse dos estudantes pela disciplina. Atividades de laboratório são essenciais para a formação do engenheiro de controle tanto no âmbito motivacional como para garantir uma melhor compreensão dos conceitos ensinados em sala de aula (Reck, 2015).

No contexto do curso de Engenharia de Controle e Automação, o estudo de controle clássico ocupa um papel central, pois fornece a base para a compreensão e desenvolvimento de sistemas automáticos. Muitas vezes, os alunos não têm a oportunidade de observar a dinâmica real de um sistema, o que os impede de entender plenamente as nuances e os desafios da aplicação prática dos controladores. Uma pesquisa de 2008 do IEEE Control Systems Society revelou que universidades superestimam a formação de seus alunos para atender às demandas da indústria. Uma parte dos representantes do setor considera a modelagem matemática essencial, e 72% apontam a experiência prática como a principal área a ser aprimorada na formação de engenheiros de controle (Cook, 2009).

Diante desse cenário, surge a questão: seria possível integrar ferramentas de software amplamente utilizadas, como MATLAB e Simulink, com hardware acessível, como o Arduino, para oferecer uma abordagem prática ao ensino de controle clássico? Essa integração permitiria não apenas que os alunos conectassem teoria e prática, mas também que desenvolvessem habilidades fundamentais para sua formação como engenheiros.

A hipótese central deste trabalho é que a aplicação de uma metodologia prática utilizando Arduino, Simulink e MATLAB pode enriquecer significativamente o aprendizado de controle clássico. Por meio de experimentos simples, como o controle de um motor DC, os alunos seriam capazes de modelar o sistema, projetar controladores e analisar a resposta dinâmica real. Essa abordagem proporcionaria uma compreensão mais profunda dos conceitos e aumentaria o interesse pela disciplina.

A escolha de um motor DC como planta experimental, juntamente com equipamentos acessíveis, como uma ponte H L298N e uma fonte de bancada, foi motivada por sua simplicidade e versatilidade. Além disso, a comunicação serial entre o Simulink e o Arduino permite o controle em tempo real, oferecendo aos alunos uma experiência prática e interativa.

Essa proposta não apenas facilita a transição entre teoria e prática, mas também prepara os estudantes para lidar com sistemas mais complexos no futuro.

Este trabalho justifica-se pela necessidade de inovar no ensino de controle clássico, tornando-o mais dinâmico e envolvente. A falta de aplicações práticas em muitas disciplinas tem sido apontada como um fator que limita o interesse dos estudantes, levando-os a perceber o conteúdo como abstrato ou desconectado da realidade. Ao introduzir uma abordagem experimental acessível, busca-se não apenas superar essas limitações, mas também inspirar os alunos a aplicarem os conceitos em projetos pessoais e profissionais.

A utilização de ferramentas acessíveis, tanto no âmbito acadêmico quanto no mercado de trabalho, é outro ponto de destaque. O Arduino, por exemplo, é amplamente utilizado devido ao seu baixo custo e flexibilidade, enquanto MATLAB e Simulink são ferramentas padrão na indústria e na pesquisa científica. Essa combinação reforça a relevância e aplicabilidade da proposta.

Espera-se que esta metodologia contribua para a formação de engenheiros mais completos, capacitados a enfrentar desafios reais com base em uma sólida compreensão teórica e prática. Além disso, ao proporcionar um ambiente de aprendizado mais envolvente, busca-se fomentar a curiosidade e o interesse pela área de controle e automação.

Por fim, ao abordar uma problemática relevante e propor uma solução prática, este trabalho visa não apenas melhorar o ensino de controle clássico, mas também servir como um modelo que pode ser adaptado para outras disciplinas e contextos. A proposta tem potencial para transformar a maneira como os estudantes percebem e aplicam os conceitos de engenharia, contribuindo para uma formação mais rica e abrangente.

Com essa abordagem, espera-se não apenas promover o aprendizado, mas também inspirar os alunos a explorarem novas possibilidades e a desenvolverem soluções criativas para problemas reais, fortalecendo sua formação e ampliando suas perspectivas profissionais.

## **1.1 Objetivos**

### **1.1.1 Objetivo Geral**

Desenvolver uma metodologia prática e acessível para o ensino de controle clássico, integrando o uso de Arduino, Simulink e MATLAB, com o objetivo de reduzir a lacuna entre teoria e prática na formação de engenheiros. A proposta visa estimular o interesse dos alunos pelas disciplinas de controle, proporcionando um aprendizado mais consistente. Isso permitirá que os futuros profissionais apliquem os conceitos de controle de forma eficaz.

### **1.1.2 Objetivos Específicos**

1. Selecionar e preparar os componentes necessários para o experimento, como Arduino Uno, motor DC com encoder, ponte H L298N e fonte de bancada;
2. Desenvolver o modelo matemático do motor DC, obtendo sua função de transferência por meio de experimentos ou análise teórica;
3. Implementar a comunicação serial entre o Arduino e o Simulink para controle em tempo real;
4. Projetar controladores clássicos, como proporcional-integral-derivativo (PID), e simular suas respostas no Simulink;
5. Validar os controladores simulados com a resposta prática do sistema experimental;
6. Analisar as discrepâncias entre os resultados teóricos e práticos e realizar ajustes no sistema para melhorar seu desempenho;
7. Comparar o desempenho do sistema controlado em malha aberta e malha fechada, destacando as vantagens da abordagem de controle fechada;
8. Avaliar o impacto da metodologia na aprendizagem dos alunos por meio de feedback qualitativo;
9. Demonstrar a viabilidade do uso de ferramentas acessíveis, como Arduino, no ensino de controle clássico;
10. Incentivar os professores a adotar essas metodologias, de forma a despertar maior interesse e motivação dos alunos pela disciplina de Controle;

## **1.2 Estrutura do Trabalho**

Este estudo está estruturado nos capítulos a seguir: o Capítulo 2 aborda o referencial teórico que fundamenta esta monografia. No Capítulo 3, são descritos os materiais e os métodos utilizados no desenvolvimento do projeto. O Capítulo 4 apresenta os resultados alcançados durante os testes realizados. Por fim, o Capítulo 5 traz as considerações finais e a conclusão do trabalho.

## **2 Referencial Teórico**

### **2.1 Introdução ao Controle**

A teoria de controle é um campo essencial para o desenvolvimento e a análise de sistemas automatizados, abrangendo áreas da engenharia e ciências. As teorias de controle comumente usadas hoje são a teoria de controle clássico (também chamada teoria de controle convencional), a teoria de controle moderno e a teoria de controle robusto. Essas teorias oferecem abordagens diferentes para a análise e o projeto de sistemas, sendo a teoria de controle moderno fundamental para lidar com sistemas complexos (Ogata, 2010).

O controle automático é aplicado em sistemas como veículos espaciais, robótica e processos industriais, sendo indispensável em operações que envolvem variáveis como temperatura, pressão e vazão. A evolução desse campo permitiu a criação de ferramentas que atendem requisitos rigorosos de precisão e confiabilidade. Muitos sistemas de controle industrial das décadas de 1940 e 1950 usavam controladores PID no controle de pressão, temperatura etc.

O desenvolvimento das teorias de controle começou com avanços na estabilidade de sistemas dinâmicos. Minorsky, em 1922, foi pioneiro ao associar a estabilidade de sistemas a equações diferenciais. Posteriormente, Nyquist desenvolveu um método para determinar a estabilidade em sistemas de malha fechada a partir de suas respostas em malha aberta. Esses trabalhos fundamentaram os métodos de resposta em frequência, como o diagrama de Bode, e o método do lugar das raízes, que dominaram a teoria de controle clássico.

Durante as décadas de 1960 e 1980, a disponibilidade de computadores digitais permitiu a análise de sistemas no domínio do tempo. Isso levou à formulação da teoria de controle moderno, que se baseia na análise do domínio do tempo em sistemas de equações diferenciais. Essa abordagem possibilitou o controle de sistemas com múltiplas entradas e saídas, mas também revelou limitações, como a sensibilidade aos erros de modelagem.

#### **2.1.1 Função de Transferência**

Denomina-se função de transferência de um sistema linear, de parâmetros concentrados, invariantes no tempo e de entrada e saída únicas a relação entre as transformadas de Laplace da variável de saída e da variável de entrada, supondo as condições iniciais nulas (Maya, 2014).

A função de transferência é uma ferramenta fundamental na análise de sistemas lineares e invariantes no tempo, amplamente utilizada em engenharia elétrica, controle automático e

processamento de sinais. Ela é definida no domínio da frequência, geralmente como uma relação algébrica entre a transformada de Laplace da saída e a entrada de um sistema, ou seja, como pode ser observado na Equação 2.1, onde  $s$  é a variável complexa no plano de Laplace. Essa função caracteriza completamente a dinâmica de um sistema linear e invariante no tempo, permitindo a descrição de como ele responde a diferentes entradas em termos de amplitude e fase, sem a necessidade de resolver diretamente equações diferenciais no domínio do tempo.

$$\frac{Y(s)}{X(s)} = \frac{b_0s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m}{a_0s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n} \quad (2.1)$$

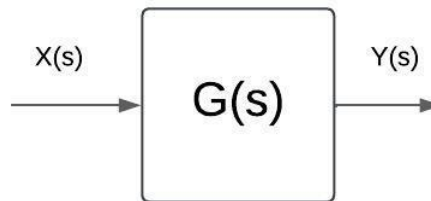
Do ponto de vista físico, a função de transferência descreve os polos e zeros de um sistema, que determinam sua estabilidade e comportamento dinâmico. Os polos, localizados no denominador, indicam as frequências naturais de oscilação e os regimes de amortecimento do sistema. Por outro lado, os zeros, localizados no numerador, correspondem às frequências onde a resposta do sistema é suprimida. A posição dos polos e zeros no plano  $s$  é crucial para a análise de estabilidade e desempenho, sendo que polos no semiplano direito indicam sistemas instáveis, enquanto aqueles no semiplano esquerdo indicam estabilidade.

Além disso, a função de transferência é amplamente utilizada no projeto e análise de controladores e filtros. No domínio da frequência, ela permite a obtenção de diagramas de Bode, Nyquist e Nichols, ferramentas gráficas que ajudam na compreensão do ganho e da fase do sistema em função da frequência. Isso facilita o ajuste de parâmetros para atingir os requisitos desejados, como estabilidade, margem de ganho e resposta transitória adequada. Assim, a função de transferência não apenas simplifica a modelagem de sistemas complexos, mas também fornece uma base sólida para o projeto de soluções eficazes em várias áreas da engenharia.

### 2.1.2 Malha Aberta

Os sistemas de controle de malha aberta são caracterizados pela ausência de realimentação, ou seja, o sinal de saída não exerce influência sobre o sistema. Nesse tipo de controle, o sinal de saída não é medido nem comparado com a entrada de referência. Um exemplo comum é o funcionamento de uma máquina de lavar roupas, em que as etapas de colocar de molho, lavar e enxaguar são executadas em sequência baseada no tempo, sem verificação de quão bem as roupas foram lavadas.

Figura 1 – Diagrama Malha Aberta



Fonte: do Autor

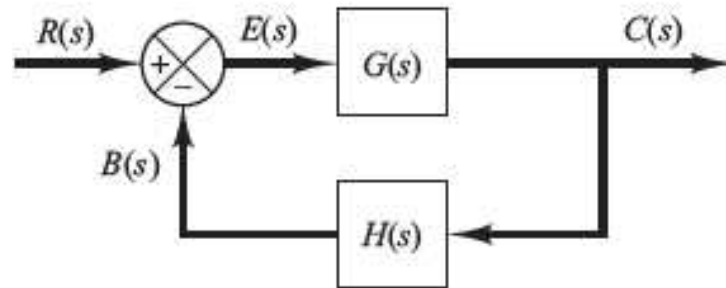
A Figura 1 ilustra um diagrama de um sistema de controle de malha aberta, evidenciando a falta de realimentação, onde. Em sistemas de controle de malha aberta, a saída não é comparada com a entrada, e cada entrada de referência corresponde a uma condição fixa de operação. Isso significa que a precisão do sistema depende de sua calibração inicial. Contudo, em situações em que ocorrem distúrbios, internos ou externos, o sistema pode falhar em realizar a tarefa esperada. Por essa razão, esses sistemas só são eficazes se a relação entre entrada e saída for bem definida e se não houver influência de distúrbios (Ogata, 2010).

Outro exemplo de sistema de controle de malha aberta é o controle de tráfego por meio de sinais temporizados. Nesse caso, a operação é baseada em um cronograma fixo de tempo, sem levar em conta as condições reais do tráfego no momento. Esses sistemas, embora úteis em certas situações, não são realimentados e, portanto, têm limitações significativas em termos de precisão e adaptabilidade.

### 2.1.3 Malha Fechada

Os sistemas de controle com realimentação, frequentemente chamados de sistemas de controle de malha fechada, utilizam o sinal de saída para ajustar continuamente o sistema em direção ao valor desejado. Nessa configuração, o termo **sinal de erro** refere-se à diferença entre o sinal de entrada e o sinal de realimentação, que pode ser o próprio sinal de saída ou uma função que inclua suas derivadas e/ou integrais. Esse sinal de erro é realimentado ao controlador para minimizar discrepâncias e ajustar a saída do sistema ao valor de referência desejado (Ogata, 2010).

Figura 2 – Diagrama Malha Fechada



Fonte: (Ogata, 2010)

A Figura 2 ilustra um diagrama de controle de malha fechada, evidenciando a interação entre a entrada e a saída.

O termo "malha fechada" implica sempre o uso de realimentação como estratégia para reduzir o erro do sistema. Esse tipo de controle é amplamente aplicado em sistemas onde a precisão e a adaptação às condições variáveis são fundamentais, sendo uma característica essencial de sistemas modernos e avançados (Ogata, 2010).

$$C(s) = G(s)E(s) \quad (2.2)$$

$$E(s) = R(s) - B(s) \quad (2.3)$$

$$E(s) = R(s) - H(s)C(s) \quad (2.4)$$

$$C(s) = G(s)[R(s) - H(s)C(s)] \quad (2.5)$$

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (2.6)$$

Conforme vemos na Equação 2.6, temos a descrição da função de transferência de um sistema de controle em malha fechada. Essa equação evidencia como a realimentação afeta o comportamento do sistema, destacando que a presença do termo no denominador incorpora o efeito da realimentação sobre a estabilidade e a resposta dinâmica. O ganho do sistema e a função de transferência da planta desempenham papéis cruciais na definição do desempenho global do sistema, sendo fundamentais no projeto de controladores que atendam aos requisitos de estabilidade e precisão.

### 2.1.4 Estabilidade

Um sistema de controle normalmente possui energia interna armazenada. Essa energia armazenada impede que a saída do sistema siga a entrada imediatamente. Conseqüentemente, quando se aplica um sinal de controle, o sistema necessita de algum tempo para responder ao estímulo. Ao atingir o regime permanente, a saída poderá conter um erro devido a esse atraso, denominado erro estacionário. Com a função de transferência do sistema, é possível, a partir de funções de teste, obter as características de resposta transitória e permanente (Dorf, 2018).

### 2.1.5 Parâmetros de Desempenho de Sistemas de 2ª Ordem

Para fins de ensino e análise preliminar do comportamento de sistemas físicos, é razoavelmente adequado modelá-los como sistemas de segunda ordem. Isso ocorre porque, em muitos casos, o comportamento dinâmico dos sistemas pode ser descrito por dois polos dominantes, que são responsáveis por definir as principais características dinâmicas do sistema.

Um sistema de segunda ordem, com retroação a malha fechada, tem seu sinal de saída modelado como (Dorf, 2018):

$$Y(s) = \frac{\omega_n^2}{s(s^2 + 2\zeta\omega_n + \omega_n^2)} \quad (2.7)$$

Como mostrado na Equação 2.7, a dinâmica de um sistema de segunda ordem é representada para uma entrada degrau unitário. A entrada degrau unitário é comumente utilizada para analisar a resposta de sistemas dinâmicos, caracterizando-se por uma mudança abrupta e constante em uma variável de controle, como um sinal de controle ou uma condição inicial.

Essa entrada é definida por um aumento repentino para um valor constante, geralmente a partir de  $t = 0$ . Ao aplicar essa entrada, é possível estudar como o sistema responde a uma alteração instantânea, revelando informações importantes sobre seu comportamento transitório, como tempo de acomodação, estabilidade, tempo de subida e sobressinal.

- $\zeta$  representa o amortecimento do sistema;
- $\omega_n$  representa a frequência natural de oscilação do sistema;

O amortecimento de um sistema está relacionado à dissipação de energia ao longo do tempo, o que faz com que a amplitude das oscilações ou flutuações do sistema diminua progressivamente até que ele atinja um estado de equilíbrio. Em sistemas dinâmicos de segunda

ordem, o amortecimento é representado por um parâmetro que controla o quão rapidamente o sistema se estabiliza após uma perturbação. Se o amortecimento for baixo (subamortecido), o sistema oscilará antes de se estabilizar. Se o amortecimento for alto (superamortecido), o sistema se estabiliza rapidamente sem oscilar. Quando o amortecimento é crítico, o sistema atinge a estabilidade no menor tempo possível, sem oscilações, mas sem um excesso de aceleração.

A frequência natural de um sistema, por outro lado, é a frequência com a qual ele tenderia a oscilar em sua forma livre, sem qualquer influência externa. Essa frequência depende das propriedades do sistema, como a massa e a rigidez, e é fundamental para determinar o comportamento dinâmico. A frequência natural está diretamente relacionada à resposta do sistema a perturbações, e, quando um sistema é excitado em sua frequência natural, ele tende a oscilar com maior amplitude. A interação entre a frequência natural e o amortecimento define o modo como o sistema se comporta após ser perturbado, incluindo a velocidade com que ele retorna ao equilíbrio e o padrão de oscilações, se houver.

As raízes da equação característica são equivalentes aos polos em malha fechada do sistema, como pode-se observar na Equação 2.8, que são:

$$s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{(\zeta^2 - 1)} \quad (2.8)$$

- $\zeta > 1$  Resposta Superamortecida com polos reais, negativos e distintos
- $0 < \zeta < 1$  Resposta Subamortecida com polos complexos conjugados
- $\zeta = 0$  Resposta sem Amortecimento com polos puramente imaginários
- $\zeta = 1$  Resposta Criticamente Amortecida com polos reais e iguais

Em aplicações práticas envolvendo problemas físicos, é comum que o cenário mais provável seja aquele em que o sistema apresenta polos complexos conjugados. Por isso, é importante destacar as medidas de desempenho específicas para essa categoria de sistemas subamortecidos. Esses sistemas, quando submetidos a uma entrada degrau unitário, exibem uma resposta característica que reflete suas propriedades dinâmicas (Nise, 2017):

$$y(t) = 1 - \frac{1}{\sqrt{(\zeta^2 - 1)}} e^{-\zeta\omega_n t} \cos \left[ \left( \omega_n \sqrt{(\zeta^2 - 1)} \right) t - \varphi \right] \quad (2.9)$$

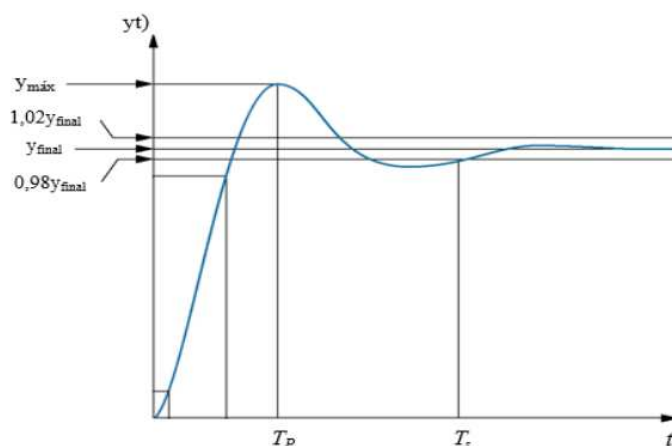
$$\varphi = \tan^{-1} \left( \frac{\zeta}{\sqrt{(\zeta^2 - 1)}} \right) \quad (2.10)$$

Vale observar que o sistema possui uma resposta transitória que pode ser observada na Equação 2.9 com oscilação senoidal amortecida. Em regime permanente, com o tempo tendendo ao infinito, não existe erro entre a entrada e a saída (Ogata, 2010).

Os índices de desempenho de um sistema desempenham um papel fundamental na avaliação e análise da qualidade de sua resposta. Eles são amplamente utilizados em estudos e aplicações práticas para quantificar o comportamento do sistema diante de diferentes condições, permitindo ajustar ou otimizar seu desempenho de forma mais eficiente. Esses índices ajudam a compreender características como estabilidade, rapidez e precisão, que são essenciais em projetos de controle.

A Figura 3 apresenta uma visão detalhada das principais medidas de desempenho empregadas especificamente para sistemas de 2ª ordem com resposta subamortecida. Esses parâmetros incluem aspectos como o tempo de subida, tempo de acomodação, tempo de pico, e o sobresinal, entre outros. Cada uma dessas medidas fornece informações valiosas sobre a dinâmica do sistema e é essencial para atender aos requisitos específicos de desempenho em aplicações práticas.

Figura 3 – Medidas de Desempenho para um Sistema Subamortecido



Fonte: (Barbosa, 2020)

- **Tempo de Acomodação:** tempo necessário para que o sistema atinja o valor de regime em torno do valor do degrau unitário. Existe um valor de tolerância de até 2% acima ou

abaixo do valor de regime, para que se considere uma variação aceita. O valor do tempo de acomodação pode ser calculado de forma simplificada por (Nise, 2017):

$$T_s = \frac{4}{\zeta\omega_n} \quad (2.11)$$

- **Tempo de Pico:** o tempo de pico pode ser compreendido como o tempo transcorrido até atingir o primeiro pico de sinal, ou valor máximo. O valor do tempo de pico pode ser calculado de forma simplificada por (Nise, 2017):

$$T_p = \frac{\pi}{\omega_n\sqrt{(1-\zeta^2)}} \quad (2.12)$$

- **Sobressinal Máximo:** O sobressinal máximo representa o quanto a forma de onda ultrapassa o valor de estado estacionário no instante de pico, sendo expresso como uma porcentagem desse valor. Em muitos casos, esse parâmetro está relacionado a questões de segurança, como a tensão máxima que um circuito pode suportar, a deformação máxima que uma estrutura pode suportar sem causar ruptura do material, entre outros. O sobressinal máximo também é conhecido como Ultrapassagem Percentual (%) e pode ser calculado conforme descrito em (Nise, 2017):

$$M_p = e^{-\frac{\zeta\pi}{\sqrt{(1-\zeta^2)}}} \cdot 100\% \quad (2.13)$$

### 2.1.6 Controladores PID

O controle refere-se ao processo de ajustar uma variável de um sistema para que ela siga uma referência previamente definida, mesmo estando sujeita a variações. Em sistemas de controle onde há uma comparação entre o sinal de saída e o sinal de entrada (referência), e essa diferença é utilizada para determinar a ação de controle, o sistema é classificado como de malha fechada ou com realimentação (Campos, 2018).

O controlador Proporcional, Integral e Derivativo (PID) é amplamente utilizado na maioria dos sistemas de controle, sendo o mais comum nas indústrias. Na literatura, há diversas abordagens para a sintonia e métodos automáticos para ajustar seus parâmetros. Esse tipo de controlador pode operar com várias malhas de controle, responsáveis por manter variáveis de processo, como vazão, temperatura e nível, dentro de valores predefinidos (Ogata, 2010)

O PID possui três parâmetros fundamentais: Ganho Proporcional ( $K_p$ ), Tempo Integral ( $T_i$ ) e Tempo Derivativo ( $T_d$ ). O parâmetro  $K_p$  representa um ganho ajustável do amplificador, enquanto  $T_i$  busca eliminar erros em regime permanente, garantindo que o erro de estado

estacionário seja nulo. Já  $T_d$  está relacionado à taxa de variação do erro, permitindo antecipar a resposta do controlador para acelerar a convergência à referência (Campos, 2018).

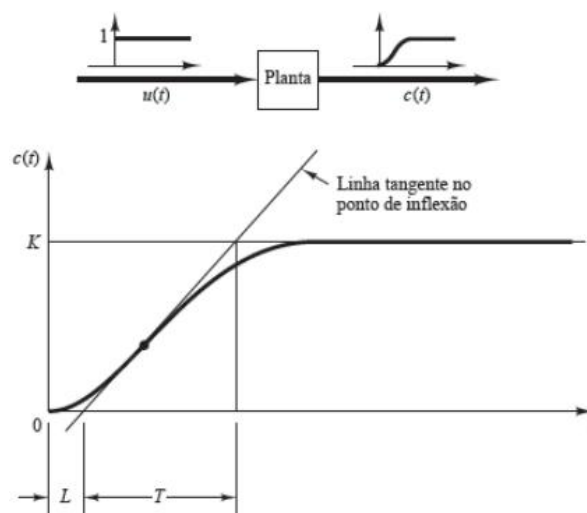
Para alcançar um bom desempenho do controlador, é essencial realizar sua sintonia. Ziegler e Nichols (1942) propuseram dois métodos clássicos para essa tarefa. O primeiro, denominado método da curva de reação, utiliza a resposta ao degrau em malha aberta. O segundo, conhecido como método do ponto crítico, ajusta o sistema em malha fechada apenas com a ação proporcional, aumentando-a gradativamente até alcançar oscilações sustentadas na saída.

### 2.1.6.1 Método da Curva de Reação

O método da curva de reação consiste em analisar a resposta dinâmica da planta em malha aberta, utilizando uma entrada em degrau  $u(t)$ . Nesse cenário, a curva do sinal de saída  $c(t)$  assume um formato em "S", desde que a planta não possua integradores nem polos conjugados complexos dominantes (Ogata, 2010).

Essa curva em "S" é definida por duas constantes principais: o tempo de atraso  $L$  e a constante de tempo  $T$ , conforme podemos observar na Figura 4. Essas constantes são determinadas traçando-se uma reta tangente ao ponto de inflexão da curva até que ela intercepte o eixo do tempo e uma linha paralela a esse eixo, representando o ponto de ganho  $K$  em regime permanente no gráfico do sinal de saída da planta (Barroso, 2019).

Figura 4 – Curva de formato S a resposta ao degrau unitário de uma planta.



Fonte: (Ogata, 2010)

Dessa forma, a função de transferência que relaciona a saída  $C(s)$  e a entrada  $U(s)$  do sistema pode ser modelada como um sistema de primeira ordem com atraso. Esse tipo de representação é amplamente utilizado para descrever o comportamento dinâmico de sistemas que apresentam uma resposta exponencial inicial com um tempo de atraso antes de começar a reagir à entrada (Ogata, 2010).

A modelagem por meio de um sistema de primeira ordem com atraso é especialmente útil em processos industriais e sistemas de controle, pois simplifica a análise e o projeto de controladores ao capturar as características essenciais do sistema. Essa abordagem pode ser expressa matematicamente pela Equação 2.14, que fornece os parâmetros necessários para compreender a dinâmica e projetar estratégias de controle apropriadas.

$$\frac{C(s)}{U(s)} = \frac{K}{Ts + 1} e^{-Ls} \quad (2.14)$$

No método da curva de reação, Ziegler e Nichols estabeleceram um procedimento prático para determinar os parâmetros do controlador com base em uma análise da resposta dinâmica do sistema em malha aberta. Esse método utiliza as características da curva de reação em formato "S" para identificar valores como o tempo de atraso  $L$  e a constante de tempo  $T$ , que são então aplicados para calcular os ganhos do controlador proporcional, integral e derivativo.

A partir dessas informações, Ziegler e Nichols propuseram uma série de regras que permitem ajustar os parâmetros do controlador de maneira sistemática, garantindo um desempenho satisfatório do sistema controlado. Essas regras foram organizadas de forma clara e objetiva, sendo apresentadas em tabelas para facilitar a sua aplicação em diferentes cenários de controle.

Dessa forma, os parâmetros do controlador para o método da curva de reação devem ser definidos conforme indicado na Tabela 1, que resume os valores recomendados para os ganhos, dependendo do tipo de ação de controle desejada. Essas recomendações fornecem uma base inicial para a sintonia do controlador, podendo ser refinadas posteriormente para atender às necessidades específicas do sistema.

Tabela 1 – Regra de sintonia de Ziegler- Nichols para o método da curva de reação.

Tipo de controlador	$K_p$	$T_i$	$T_d$
<b>P</b>	$\frac{T}{L}$	$\infty$	0
<b>PI</b>	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
<b>PID</b>	$1,2 \frac{T}{L}$	$2L$	$0,5L$

Fonte: (Ogata, 2010)

### 2.1.6.2 Controlador Proporcional

Em um controlador que utiliza exclusivamente a ação proporcional, a relação entre o sinal de controle  $u(t)$  e o sinal de erro  $e(t)$  é descrita pela Equação 2.15. Esse tipo de controlador é caracterizado pela simplicidade e eficiência em situações onde o sistema exige uma resposta rápida, já que o sinal de controle é diretamente proporcional ao erro presente entre a saída e o valor desejado. No entanto, um aspecto importante a ser destacado é que o controlador proporcional, quando operando com um sinal de referência fixo, tende a apresentar um erro de regime permanente. Isso significa que, após um tempo de estabilização, o sistema não conseguirá corrigir totalmente o erro, permanecendo com uma diferença residual entre a saída e o valor de referência (Campos, 2018)

Uma das principais características desse erro de regime permanente é que sua magnitude diminui à medida que o ganho  $K_p$  do controlador é aumentado. Ou seja, quanto maior for o valor de  $K_p$ , menor será o erro de regime permanente. Contudo, é importante observar que um aumento excessivo no ganho  $K_p$  pode levar a uma resposta mais instável ou a oscilações indesejadas no sistema. Dessa forma, o ajuste adequado do ganho  $K_p$  é fundamental para otimizar a performance do controlador, balanceando a redução do erro com a estabilidade do sistema. Esse tipo de comportamento é amplamente discutido em sistemas de controle clássico e é um fator crucial na escolha do tipo de controlador a ser utilizado em diversas aplicações práticas (Franklin, 2013).

$$u(t) = K_p e(t) \quad (2.15)$$

Com isso, a função de transferência do Controlador Proporcional (P) no domínio de Laplace é expressa na Equação 2.16, representando a relação entre a entrada e a saída do sistema de controle. Esse modelo matemático descreve como o controlador P reage a uma variação na

entrada, fornecendo uma saída proporcional à amplitude do erro, sem considerar termos derivados ou integradores, o que simplifica sua implementação e análise. A Equação 2.16 ilustra claramente esse comportamento, evidenciando a eficiência e a simplicidade desse tipo de controlador.

$$\frac{U(s)}{E(s)} = K_p \quad (2.16)$$

### 2.1.6.3 Controlador Proporcional Integral

Em um controlador que combina a ação proporcional com a integral, a relação entre o sinal de controle  $u(t)$  e o sinal de erro  $e(t)$  é dada pela Equação 2.17. A inclusão do termo integral no controlador ajuda a superar a limitação do controlador  $P$ , pois a saída desse termo é um somatório dos valores anteriores de  $e(t)$ . Isso permite corrigir o erro de regime permanente, contribuindo para uma resposta mais precisa e eficiente do sistema (Barroso, 2019).

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \quad (2.17)$$

Dessa forma, a função de transferência do Controlador Proporcional-Integral ( $PI$ ) no domínio de Laplace é expressa na Equação 2.18. Esse controlador combina as ações proporcional e integral, proporcionando uma resposta mais eficaz ao sistema em comparação com o controlador proporcional isolado. A parte proporcional responde diretamente ao erro atual, enquanto a parte integral corrige o erro acumulado ao longo do tempo, eliminando o erro de regime permanente. A Equação 2.18 ilustra matematicamente como essas duas ações interagem no domínio de Laplace, resultando em um controle mais preciso.

$$\frac{U(s)}{E(s)} = K_p \left( 1 + \frac{1}{T_i s} \right) \quad (2.18)$$

### 2.1.6.4 Controlador Proporcional Integral Derivativo

O controlador que incorpora as ações proporcional, integral e derivativa é uma combinação do controlador  $PI$  com o termo derivativo. Dessa forma, a relação entre o sinal de controle  $u(t)$  e o sinal de erro  $e(t)$  é apresentada na Equação 2.19. A ação do termo derivativo depende da taxa de variação do erro, o que permite antecipar a correção do sinal de erro. Esse

comportamento contribui para a redução do tempo de resposta e para a melhoria da estabilidade do sistema, tornando o controle mais eficiente e dinâmico (Campos, 2018).

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (2.19)$$

Dessa forma, a função de transferência do Controlador Proporcional-Integral-Derivativo (*PID*) no domínio de Laplace é apresentada na Equação 2.20. Este controlador combina as três ações fundamentais que são a proporcional, integral e derivativa, permitindo uma resposta mais robusta ao sistema.

A parte proporcional responde ao erro atual, a parte integral corrige o erro acumulado ao longo do tempo, enquanto a parte derivativa antecipa a mudança no erro, ajudando a minimizar oscilações e melhorar a estabilidade. A Equação 2.20 descreve matematicamente como essas três ações interagem no domínio de Laplace, proporcionando um controle mais eficiente e preciso, adequado para sistemas dinâmicos complexos.

$$\frac{U(s)}{E(s)} = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (2.20)$$

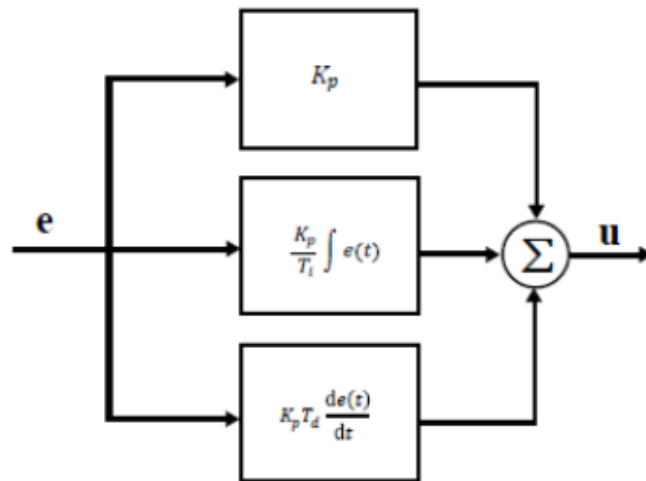
Nas Figuras 5 e 6, estão ilustradas as formas *PID* paralela ideal e *PID* série, respectivamente. O controlador *PID* é implementado em sua maioria das vezes na forma paralela ideal, que pode ser apresentado pelo Ganho Integral e Derivativo, os quais se relacionam com Ganho Proporcional conforme as Equações 2.21 e 2.22, respectivamente.

$$K_i = \frac{K_p}{T_i} \quad (2.21)$$

$$K_d = K_p T_p \quad (2.22)$$

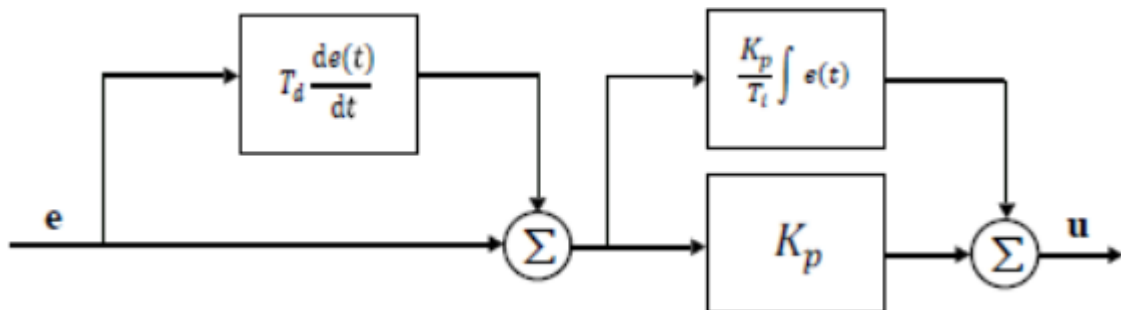
A Figura 7 ilustra um sistema de malha fechada com um controlador *PID*, onde pode-se observar a estrutura geral de uma malha de controle. O formato da saída do sistema depende de que forma o controlador foi projetado para um determinado fim, assim, é de competência do Engenheiro de Controle usar seu conhecimento para projetar controladores que atendam os requisitos propostos.

Figura 5 – Diagrama de Blocos Controlador PID Paralelo



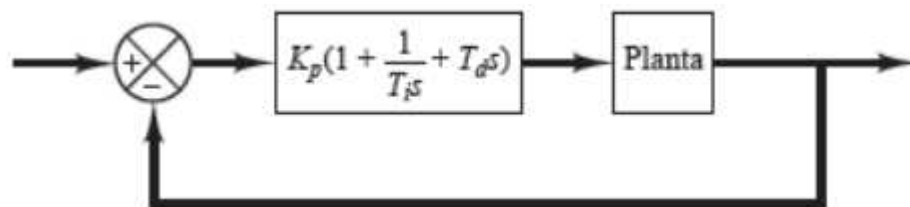
Fonte: (Pinto, 2014)

Figura 6 – Diagrama de Blocos Controlador PID Série



Fonte: (Pinto, 2014)

Figura 7 – Diagrama de Blocos Controlador PID de uma Planta Genérica



Fonte: (Pinto, 2010)

## 2.2 Ensino de Controle Clássico

Ao se abordar o ensino de controle clássico, um aspecto crucial a ser destacado, pois impacta diretamente a formação do engenheiro, é a compreensão de que, na sociedade atual, o controle é cada vez mais onipresente, desempenhando um papel central em diversas áreas de

atuação e incorporando complexidades crescentes. O controle clássico refere-se ao conjunto de técnicas tradicionais de controle de sistemas dinâmicos, em sua maioria baseadas em abordagens analógicas, como os controladores Proporcional, Integral e Derivativo (PID), que são usados para garantir que o comportamento de um sistema esteja dentro dos parâmetros desejados.

A tecnologia dos sistemas de controle, dentro dessa abordagem, é considerada um pilar essencial na revolução da automação, com impactos significativos em setores como a eletrônica, a indústria e os sistemas de produção. Isso coloca uma responsabilidade sobre a educação em controle, pois é fundamental que os futuros engenheiros adquiram uma visão abrangente, capaz de lidar com sistemas e tecnologias tradicionais de controle. Essas tecnologias são aplicáveis em contextos multidisciplinares e essenciais para o desenvolvimento de tecnologias em constante evolução (Murray, 2002).

Essa visão de ensino no controle clássico permite que os engenheiros se tornem aptos a trabalhar com os desafios da automação industrial e da regulação de sistemas dinâmicos, utilizando abordagens fundamentais, como os controladores PID e outras técnicas consagradas. Essas abordagens são amplamente aplicadas em sistemas industriais e de controle, proporcionando uma base sólida para enfrentar os desafios da engenharia moderna e contribuindo para o avanço das tecnologias de controle em várias áreas de aplicação.

### **2.2.1 Importância do Ensino Prático em Engenharia de Controle**

A importância do ensino prático na Engenharia de Controle tem se destacado ao longo do tempo, especialmente com a inserção das Tecnologias da Informação e Comunicação (TIC) no processo educacional. A partir de 1997, políticas públicas priorizaram a formação de "laboratórios de informática" nas escolas, com a implementação de tecnologias como computadores, tablets e lousas digitais (Lucena, 2016).

Nesse cenário, a tecnologia oferece oportunidades significativas, como os laboratórios virtuais, que podem representar conceitos essenciais para cursos de engenharia, incluindo a área de controle automático. Laboratórios virtuais proporcionam benefícios como acessibilidade, redução de custos e segurança, permitindo a realização de experimentos que seriam perigosos ou inviáveis em ambientes físicos.

A engenharia de controle tem se consolidado ao longo dos últimos 60 anos, especialmente pela sua aplicabilidade em diversas áreas, como eletrônica, sistemas de produção, e até mesmo na medicina. Experimentações laboratoriais são fundamentais para a educação em engenharia, pois proporcionam uma compreensão mais profunda e prática dos

sistemas, essencial para que os futuros engenheiros possam integrar rapidamente seus conhecimentos teóricos em sistemas reais (Heradio, 2016).

A demanda crescente por sistemas mais complexos exige que os engenheiros desenvolvam habilidades multidisciplinares, como controle, eletrônica e computação. O ensino baseado em projetos (PBL) é um método pedagógico que estimula o aprendizado prático e prepara os alunos para o mercado de trabalho ao integrar teoria e prática de forma eficaz.

A modelagem de sistemas físicos e controladores PID são fundamentais para a formação dos engenheiros de controle. Essa abordagem, apesar de não cobrir todas as características físicas dos sistemas, é essencial para a análise e controle dinâmico, representando a base de muitas aplicações industriais. A rápida evolução tecnológica e a crescente demanda da indústria por profissionais qualificados reforçam a necessidade de um ensino que conecte teoria e prática, para atender às exigências do mercado de trabalho.

### **2.2.2 Desafios no Ensino de Controle Clássico**

Os desafios no ensino de controle clássico estão diretamente ligados à necessidade de estabelecer uma base sólida que permita aos futuros engenheiros lidar com os problemas emergentes e crescentes da engenharia de controle. A educação em controle deve proporcionar um aprendizado contínuo, que envolva tanto o domínio dos conceitos fundamentais quanto a capacidade de aplicar esses conceitos na prática. Isso envolve o desenvolvimento de uma compreensão aprofundada de aspectos chave como: a dinâmica dos sistemas, a estabilidade, a realimentação e a compensação dinâmica. Cada um desses pontos é essencial para garantir que os engenheiros de controle possam atuar de forma eficaz em sistemas cada vez mais complexos e multidisciplinares.

Uma das principais questões pedagógicas que surgem nesse contexto é o equilíbrio entre a base conceitual e os procedimentos experimentais. A natureza do controle clássico, que combina aspectos matemáticos e físicos, apresenta um desafio único para a educação. O controle envolve tanto conceitos abstratos, como estabilidade e realimentação, quanto a manipulação de sistemas reais e materiais. Esse equilíbrio entre teoria e prática torna-se um aspecto central, pois a integração de ambos é crucial para a formação completa do engenheiro de controle. Os experimentos laboratoriais não devem ser vistos apenas como atividades complementares, mas como uma parte fundamental do processo de aprendizagem, permitindo que os alunos integrem e vivenciem os conceitos adquiridos (Ferreira, 2006).

Ademais, é amplamente reconhecido que a eficácia do ensino melhora quando múltiplos sentidos são envolvidos no processo de aprendizagem. A assimilação de conceitos abstratos é

muito mais eficaz quando acompanhada de experiências práticas, visualizações e explicações auditivas. Esse aspecto se torna ainda mais relevante no ensino de controle clássico, onde os conceitos fundamentais, como a dinâmica de sistemas, a estabilidade e a compensação dinâmica, exigem uma abordagem integrada entre teoria e prática. Assim, para enfrentar esses desafios, é essencial que os cursos de controle clássico adotem uma estrutura de ensino que combine ensino teórico com experimentos práticos, motivando os alunos a aplicar os conceitos abstratos a problemas práticos, onde a teoria se torna a solução para desafios experimentais.

### **2.2.3 Ferramentas Tradicionais de Ensino de Controle**

No ensino de controle clássico, uma das ferramentas mais tradicionais e amplamente utilizadas são os modelos matemáticos de sistemas dinâmicos. Estes modelos desempenham um papel fundamental na compreensão da dinâmica dos sistemas, pois fornecem uma descrição precisa e quantitativa da relação entre as entradas e as saídas dos sistemas de controle. Através deles, é possível representar e analisar a evolução temporal das variáveis envolvidas no processo controlado.

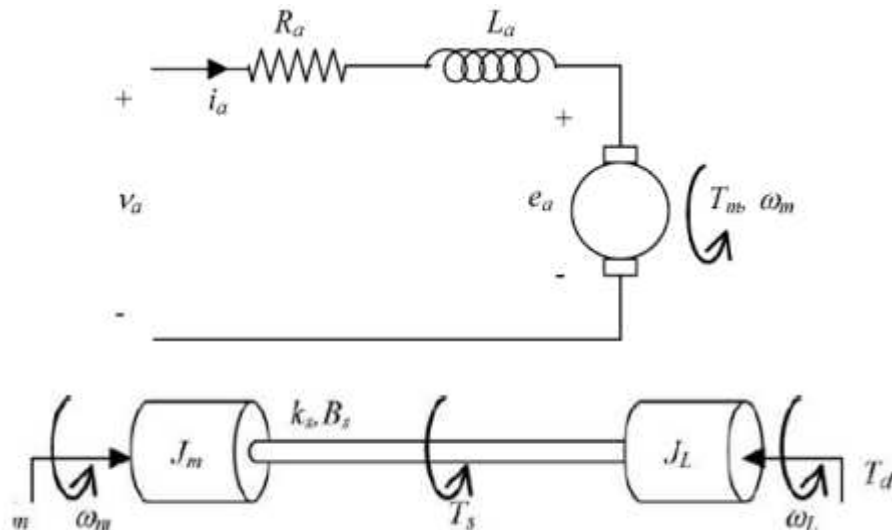
A vantagem de usar modelos matemáticos na educação em controle está na capacidade de formalizar o comportamento do sistema de maneira que permita prever e entender sua resposta a diferentes condições de operação. Isso é especialmente útil quando se trata de sistemas complexos, onde uma abordagem puramente experimental ou empírica não seria suficiente para compreender a dinâmica do sistema em questão. Além disso, os modelos matemáticos permitem a aplicação de diversas técnicas analíticas e computacionais, como a análise de estabilidade, a linearização de sistemas não-lineares e o uso de transformadas de Laplace, para analisar a resposta do sistema a entradas específicas, como degraus, rampas ou perturbações (Aguirre, 2012).

Esses modelos matemáticos também facilitam a implementação de controladores e ajudam os estudantes a entenderem como os parâmetros do controlador afetam o desempenho do sistema. Por exemplo, um controlador PID (Proporcional, Integral e Derivativo) pode ser projetado a partir do modelo matemático de um motor para regular sua velocidade de forma precisa, compensando variações na carga ou na alimentação. A capacidade de manipular e modificar os parâmetros do controlador dentro de um modelo matemático fornece aos alunos uma compreensão mais profunda dos efeitos das diferentes estratégias de controle sobre o comportamento do sistema, promovendo uma aprendizagem mais eficaz.

Através do estudo dos modelos matemáticos, os alunos são capacitados não apenas a entender o comportamento dos sistemas de controle, mas também a projetar e otimizar sistemas

de controle para uma ampla gama de aplicações, como o controle de processos industriais, robótica, automação e sistemas de transporte. Dessa forma, os modelos matemáticos servem como a base para o desenvolvimento de habilidades essenciais para a prática profissional em engenharia de controle.

Figura 8 – Exemplo de um Sistema Elétrico e Mecânico de um Motor CC

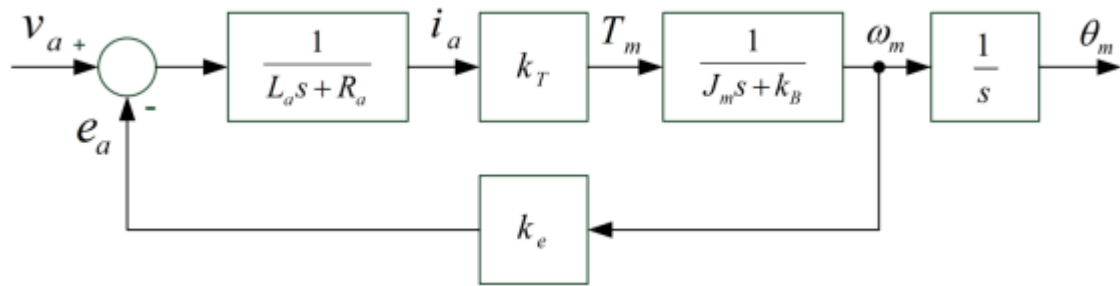


Fonte: (Aguirre, 2012)

A Figura 8 apresenta um diagrama típico de um sistema dinâmico, cujas equações podem ser derivadas a partir da aplicação das leis físicas que governam o sistema. Este diagrama permite uma análise aprofundada do comportamento do sistema, abrangendo aspectos cruciais como a estabilidade e os efeitos das variações nos parâmetros do sistema. A partir desse modelo, conceitos fundamentais, como a estabilidade do sistema e sua resposta a perturbações externas, podem ser explorados de forma mais detalhada. Ferramentas analíticas, como o lugar das raízes e o diagrama de Bode, são essenciais para avaliar o desempenho do sistema em diferentes condições e entender como as mudanças nos parâmetros influenciam o comportamento geral.

Além dos modelos matemáticos, as simulações computacionais desempenham um papel crucial no ensino de controle, pois proporcionam uma maneira prática e interativa de compreender e analisar sistemas dinâmicos.

Figura 9 – Exemplo de um Diagrama de Blocos de um Motor CC



Fonte: (Aguirre, 2012)

A Figura 9 mostra o diagrama de blocos do sistema, que relaciona a posição angular com a tensão aplicada. A partir da simulação que pode ser feita via softwares, os alunos podem observar diretamente o comportamento do sistema, analisando gráficos de resposta ao degrau e resposta em frequência. Esse tipo de ferramenta facilita a compreensão das diferenças entre teoria e prática, ajudando a consolidar os conceitos aprendidos em sala de aula de forma mais visual e experimental.

Essas ferramentas tradicionais, como os modelos matemáticos e as simulações computacionais, são essenciais para a formação de engenheiros capacitados, pois permitem uma compreensão mais profunda dos sistemas de controle e facilitam a aplicação prática dos conceitos teóricos.

### 2.3 Arduino

O Arduino é uma plataforma de prototipagem eletrônica amplamente reconhecida por sua versatilidade e acessibilidade. Consistindo em um microcontrolador e um ambiente de programação intuitivo, ele permite que usuários de diversos níveis de experiência desenvolvam projetos interativos e automatizados. Sua simplicidade de uso, combinada com uma vasta documentação e uma comunidade ativa, torna o Arduino uma ferramenta indispensável para quem deseja explorar o mundo da eletrônica e programação (Cultura, 2010).

Uma das principais vantagens do Arduino é a oferta de entradas e saídas analógicas e digitais, que permitem a conexão direta com sensores, atuadores e diversos periféricos. Com isso, é possível monitorar variáveis do ambiente, como temperatura e luz, além de controlar dispositivos como motores e LEDs. Essa integração facilita o desenvolvimento de sistemas personalizados, atendendo às necessidades de projetos em áreas como automação residencial, monitoramento ambiental e protótipos industriais.

Por ser uma solução acessível e intuitiva, o Arduino é especialmente popular entre iniciantes e entusiastas de tecnologia. Ele possibilita a criação de dispositivos eletrônicos complexos com relativa facilidade, como robôs educacionais e sistemas de controle automatizado. Além disso, sua aplicação vai além do aprendizado básico, sendo amplamente utilizado em projetos acadêmicos, experimentos científicos e até mesmo em startups inovadoras. Essa abrangência faz do Arduino uma ponte entre teoria e prática no desenvolvimento de tecnologia.

### **2.3.1 Definição e Principais Conceitos**

O Arduino é uma placa de prototipagem originária da Itália, concebida para facilitar o desenvolvimento de projetos que requerem as funcionalidades de um microcontrolador. Embora tenha tido origem no âmbito educacional, seu lançamento resultou em um sucesso tão amplo que extrapolou as fronteiras acadêmicas, conquistando também o cenário empresarial e doméstico com uma série de inovações notáveis.

Com o auxílio do Arduino, torna-se possível a captação de sinais provenientes de sensores, internet, SMS, entre outros, com subsequente conversão em sinais de saída destinados a acionar dispositivos diversos, como motores, lâmpadas, televisores etc.

### **2.3.2 Histórico**

O Arduino teve sua origem como um projeto concebido pelo professor Massimo Banzi, que o desenvolveu no Interaction Design Institute (Instituto de Design de Interação) em Ivrea, na Itália. No ano de 2005, juntamente com seus colegas de trabalho, Banzi concebeu a ideia de criar uma placa de prototipagem eletrônica própria. O objetivo era possibilitar atividades educacionais com um custo substancialmente menor em comparação aos sistemas similares disponíveis naquela época. A equipe elaborou uma plataforma que permitia aos estudantes criar projetos de eletrônica de maneira descomplicada, aproveitando microcontroladores acessíveis (Cultura, 2010).

### **2.3.3 Vantagens**

As vantagens deste microcontrolador incluem seu baixo custo e a capacidade de expandir a placa por meio de módulos que podem ser conectados à placa principal (Arduino) e são chamados de Shields. Uma característica distintiva significativa do Arduino é a sua abordagem open-source. Tanto os esquemas das placas quanto os códigos de aplicativos são

disponibilizados de forma gratuita na internet, permitindo que sejam modificados, compartilhados, expandidos e refinados por qualquer indivíduo.

Devido à sua facilidade de uso, acompanhada de um amplo suporte para programadores novatos, além das melhorias constantes, a plataforma Arduino se estabeleceu como uma ferramenta excepcional para projetos no nível inicial.

#### 2.3.4 *Hardware*

Um dos modelos de Arduino mais utilizados é o Uno que se encontra na Figura 10 como pode-se observar.

Figura 10 – Arduino Uno



Fonte: (Parts-Express, 2023)

O Arduino Uno é uma placa de prototipagem eletrônica de código aberto amplamente utilizada em projetos de eletrônica e automação. Desenvolvido para ser acessível e fácil de usar, é uma excelente ferramenta tanto para iniciantes quanto para profissionais. Seu custo é baixo, e a plataforma oferece expansibilidade através de módulos chamados Shields, que permitem adicionar funcionalidades como comunicação sem fio, controle de motores e sensores.

Além disso, a placa e os códigos são disponibilizados de forma gratuita, o que possibilita modificações e adaptações para diferentes aplicações. O Arduino Uno é suportado por uma vasta comunidade, que oferece tutoriais, fóruns e bibliotecas, tornando-o ainda mais acessível. Sua simplicidade, juntamente com a flexibilidade para criar projetos inovadores, o torna ideal para quem está começando na eletrônica ou para projetos de prototipagem rápida e desenvolvimento de novos dispositivos.

Tabela 2 – Especificações Técnicas do Arduino

<b>ESPECIFICAÇÕES</b>	
<b>Microcontrolador</b>	ATmega328
<b>Velocidade de Clock</b>	16MHz
<b>Corrente DC por I/O</b>	40mA
<b>Corrente DC por pino 3,3V</b>	50mA
<b>Pinos analógicos de Entrada</b>	6
<b>Pinos de I/O Digital</b>	14 (dos quais 6 fornecem saída PWM)
<b>Tensão de alimentação (recomendado)</b>	7 a 12 Volts
<b>Tensão de alimentação (limites)</b>	6 a 20 Volts
<b>Tensão de operação (nível lógico)</b>	5 Volts

Fonte: (Neri, 2014)

No geral, suas especificações técnicas variam entre diferentes modelos, na Tabela 2 podemos ver para o Arduino Uno, mas aqui estão algumas características gerais que são comuns à maioria das placas Arduino:

- **Microcontrolador:** A placa Arduino é equipada com um microcontrolador, geralmente da família Atmel AVR ou outros microcontroladores compatíveis. O microcontrolador é o cérebro da placa e executa os programas carregados.
- **Clock:** O microcontrolador possui um clock interno que determina a taxa de operação do processador. A velocidade do clock pode variar entre diferentes modelos de Arduino.
- **Memória:** As placas Arduino incorporam uma variedade de tipos de memória essenciais para seu funcionamento. A memória Flash é usada para armazenar o programa ou código que é carregado na placa, fornecendo as instruções necessárias para a execução das tarefas programadas. A SRAM, por sua vez, é uma memória volátil que desempenha um papel crucial ao armazenar variáveis e dados temporários durante a operação do programa, permitindo o processamento eficiente e em tempo real. Complementando essas, a memória EEPROM é uma forma de memória não volátil, especialmente utilizada para armazenar dados persistentes, que permanecem intactos mesmo quando a placa é reiniciada. Essa diversidade de memórias permite que as placas Arduino executem suas tarefas de maneira eficaz, manipulando programas, dados temporários e informações permanentes de forma coordenada.

- Pinos de E/S: As placas Arduino têm uma variedade de pinos de entrada/saída digital e analógica que podem ser usados para conectar sensores, atuadores e outros dispositivos externos. Esses pinos podem ser configurados como entradas ou saídas.
- Portas Analógicas: Placas Arduino geralmente têm portas analógicas que permitem a leitura de sinais analógicos, como sensores de luz ou temperatura.
- Conectividade: Muitas placas Arduino possuem interfaces para comunicação, como USB, UART (Serial), I2C e SPI, permitindo a interação com outros dispositivos ou a comunicação com um computador.
- Alimentação: A maioria das placas Arduino pode ser alimentada por meio de uma porta USB ou uma fonte de energia externa. A faixa de tensão de operação pode variar, mas geralmente é 5V ou 3.3V.
- Reguladores de Tensão: As placas frequentemente possuem reguladores de tensão integrados para fornecer voltagens estáveis para o microcontrolador e outros componentes.
- LEDs de Indicação: Placas Arduino normalmente possuem LEDs de indicação, como o LED de alimentação e LEDs de status que podem ser controlados pelo usuário.
- Botões de Reset e Programação: A maioria das placas tem um botão de reset para reiniciar o programa e uma maneira de colocá-las em modo de programação para carregar novo código.

### 2.3.5 *Software*

Uma das principais vantagens do Arduino é a sua acessibilidade, uma vez que não exige licenças de desenvolvedor para uso. Além disso, ele é acompanhado por um software específico, a IDE (Ambiente Integrado de Desenvolvimento), que permite a programação utilizando uma linguagem de alto nível conhecida como "Linguagem de Programação do Arduino", fundamentada no C++.

A comunicação entre o Arduino e o computador é estabelecida por meio de um cabo USB, o que o torna acessível para uma ampla gama de pessoas com conhecimentos medianos em informática e eletrônica.

#### 2.3.5.1 IDE

É o ambiente de programação concebido pelos mesmos criadores da placa. Foi desenvolvido em colaboração com um dos criadores do programa Processing, também criado

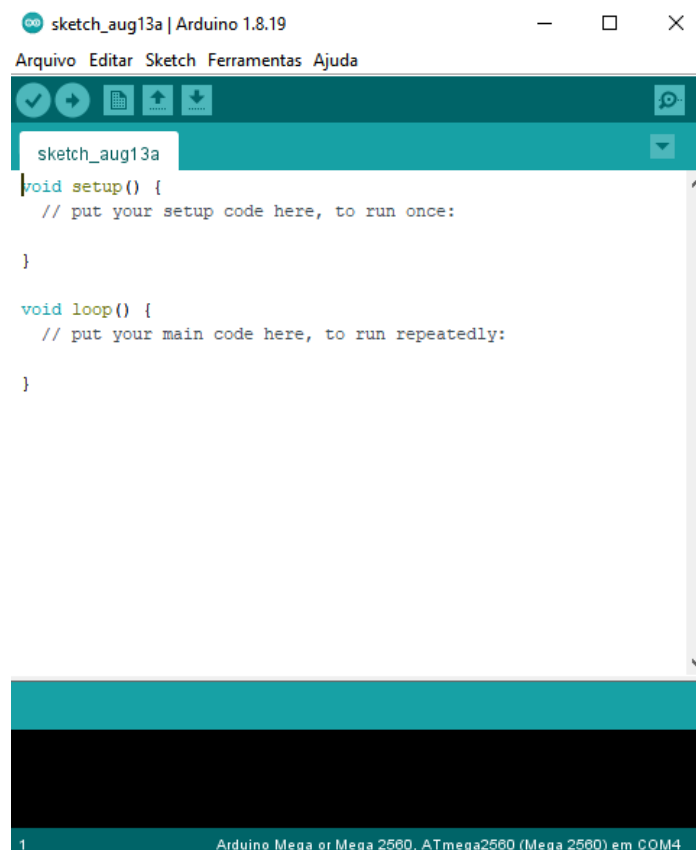
no MIT. Como resultado, ambos compartilham essencialmente a mesma interface e aderem ao mesmo paradigma, que envolve uma função de configuração seguida por um loop infinito. Esse ciclo contínuo se alinha bem com a finalidade do microcontrolador, que varia desde o controle de produtos ou periféricos até a automação de uma ou várias tarefas simultaneamente.

Em ambos os contextos de aplicação, é fundamental programar o controlador para executar a mesma tarefa repetidamente, quantas vezes forem necessárias, o que nem sempre é previsível.

Todo programa destinado ao Arduino deve incluir obrigatoriamente as duas funções principais: `setup()` e `loop()`. A primeira é executada imediatamente após a placa ser ligada, antes de qualquer outra função. Portanto, ela é comumente utilizada para configurar os pinos de entrada/saída da placa e declarar variáveis.

A segunda função, que é executada imediatamente após o `setup()`, consiste em um loop infinito, isto é, o bloco de comandos reinicia continuamente quando termina. Curiosamente, é possível invocar uma dessas funções dentro do escopo da outra.

Figura 11 – IDE Arduino



```
sketch_aug13a | Arduino 1.8.19
Arquivo Editar Sketch Ferramentas Ajuda
sketch_aug13a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

1 Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) em COM4
```

Fonte: do Autor

O fabricante do Arduino oferece gratuitamente o Arduino IDE (Interface de Desenvolvimento Integrado, conforme ilustrado na Figura 11), que é o software utilizado para programar a placa, baseado na linguagem C++. Com essa ferramenta, é possível selecionar qual versão específica do Arduino será programada, além de disponibilizar bibliotecas que simplificam o processo de elaboração do programa.

## 2.4 Matlab e Simulink

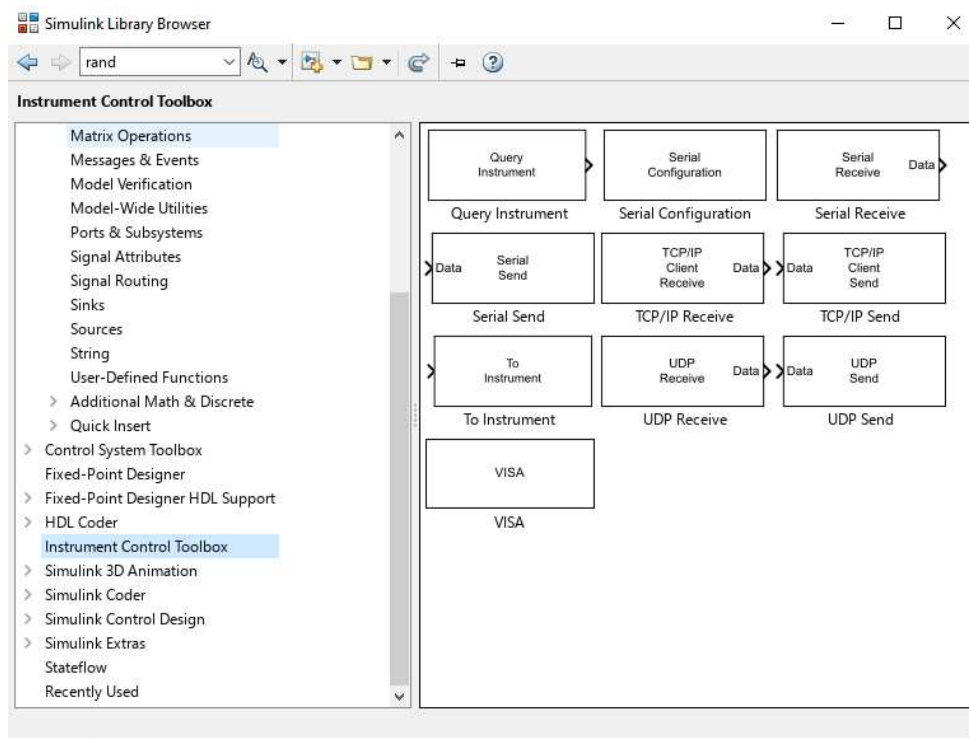
O MATLAB é uma poderosa plataforma de computação numérica, visualização e programação, amplamente reconhecida no meio acadêmico e industrial, desenvolvida pela MathWorks, Inc. Ele combina uma linguagem de alto nível com um ambiente interativo que facilita a exploração de dados, o desenvolvimento de algoritmos e a criação de soluções computacionais complexas (Abrantes, 2019).

Nos últimos anos, o MATLAB consolidou sua posição como uma ferramenta de referência em diversas áreas, como computação numérica, processamento de sinais, análise de dados e desenvolvimento de sistemas dinâmicos. Originalmente, o software estava disponível apenas em sua versão profissional, voltada para pesquisadores e profissionais. No entanto, o lançamento de uma versão mais acessível para estudantes expandiu significativamente seu uso, tornando-o uma ferramenta indispensável em salas de aula e laboratórios de ensino.

O ecossistema MATLAB inclui o programa principal e uma vasta gama de *toolboxes* — coleções de bibliotecas especializadas que estendem suas funcionalidades para atender necessidades específicas em áreas como controle, processamento de imagens, aprendizado de máquina e modelagem de sistemas físicos, como pode-se observar na Figura 12. Entre esses, destaca-se o Control System Toolbox, que, combinado com o programa principal, oferece um conjunto robusto de recursos para o projeto, análise e simulação de sistemas de controle, atendendo tanto às demandas educacionais quanto às industriais.

Além disso, o MATLAB vem se integrando cada vez mais a outras tecnologias e linguagens de programação, como Python, C++ e sistemas baseados em nuvem, o que permite maior interoperabilidade e facilita a incorporação de fluxos de trabalho complexos. Essa capacidade de integração tem sido essencial para atender às demandas crescentes por soluções híbridas e personalizáveis em ambientes acadêmicos e industriais. Como resultado, o MATLAB tem se tornado uma peça central em projetos colaborativos e multidisciplinares, onde a necessidade de combinar diferentes ferramentas, dados e métodos é fundamental para enfrentar os desafios tecnológicos e científicos contemporâneos de maneira eficiente e inovadora.

Figura 12 – Bibliotecas Simulink



Fonte: do Autor

### 2.4.1 Integração do Matlab com Simulink

O Simulink é uma plataforma integrada ao MATLAB, permitindo uma interação fluida entre ambos os ambientes. Ele é capaz de incorporar algoritmos desenvolvidos no MATLAB diretamente em seus modelos, o que viabiliza o uso de códigos personalizados e ferramentas avançadas de análise. Além disso, os resultados das simulações realizadas no Simulink podem ser facilmente exportados para o MATLAB, possibilitando análises detalhadas e pós-processamento dos dados simulados (Abrantes, 2019).

Como um ambiente de simulação baseado em diagramas de blocos, o Simulink é amplamente utilizado para projetar, modelar e simular sistemas dinâmicos. Ele suporta atividades como design em nível de sistema, simulação, geração automática de código e verificação contínua de sistemas embarcados. Essa integração robusta e funcionalidade abrangente tornam o Simulink uma ferramenta indispensável para engenheiros e pesquisadores que trabalham com projetos complexos.

### 2.4.2 Aplicações e Recursos do Simulink

A utilização do Simulink tem se destacado especialmente no desenvolvimento de sistemas de controle avançados, sendo reconhecida como uma solução eficiente para otimizar

tempo e reduzir custos de projeto. Sua arquitetura aberta proporciona flexibilidade aos usuários, permitindo a criação de bibliotecas personalizadas de blocos que podem ser reutilizadas em projetos futuros ou compartilhadas para apoiar pesquisas científicas (Skraba, 2020).

Além de sua versatilidade, o Simulink oferece suporte a uma ampla gama de pacotes adicionais, que podem ser instalados conforme necessário para atender a aplicações específicas. Esses pacotes incluem bibliotecas voltadas para áreas como controle de sistemas, processamento de sinais, modelagem física e integração com hardware.

No contexto deste trabalho, foram utilizados os pacotes de suporte do MATLAB e do Simulink para o Arduino, demonstrando sua aplicabilidade na integração de software com hardware. Essa abordagem não apenas amplia as possibilidades de modelagem e simulação, mas também facilita a transição para testes em sistemas reais, conectando teoria e prática de forma eficiente.

### **2.4.3 Simulink Coder**

O Simulink Coder é uma extensão avançada do Simulink que possibilita a geração automática de código em linguagens como C e C++ a partir de modelos baseados em diagramas de blocos. Desenvolvido pela MathWorks, esse recurso permite não apenas criar código eficiente e otimizado, mas também carregá-lo diretamente para plataformas de hardware selecionadas, como microcontroladores, sistemas embarcados e dispositivos IoT.

Essa funcionalidade simplifica significativamente o processo de desenvolvimento, eliminando etapas manuais de programação e reduzindo potenciais erros. Dessa forma, o Simulink Coder acelera o ciclo de desenvolvimento, especialmente em projetos complexos que exigem integração de software e hardware (Lobo, 2017).

#### **2.4.3.1 Principais Recursos do Simulink Coder**

O Simulink Coder, em conjunto com as ferramentas e componentes da MathWorks, oferece uma ampla gama de funcionalidades que tornam o processo de design e implementação mais eficiente e flexível. Entre seus principais recursos, destacam-se:

- **Geração de Código Automático:** Facilita a criação de código para diferentes plataformas, como microcontroladores, sistemas embarcados e computadores, otimizando a portabilidade e a eficiência do desenvolvimento.
- **Desenvolvimento Rápido e Direto:** Permite implementar sistemas diretamente a partir dos modelos criados no Simulink, acelerando o ciclo de prototipagem e validação.

- **Integração MATLAB-Simulink:** Oferece suporte nativo para a integração com o MATLAB, ampliando as possibilidades de análise e personalização.
- **Interface Gráfica Intuitiva:** Proporciona uma interface amigável que simplifica a configuração e operação do processo de geração de código, mesmo para usuários iniciantes.
- **Arquitetura Aberta e Extensível:** Possui uma estrutura flexível que permite a implementação de novas extensões e personalizações, atendendo às demandas específicas de diferentes projetos e indústrias.

#### **2.4.3.2 Aplicações e Benefícios**

O Simulink Coder tem sido amplamente utilizado em áreas como automação, aeroespacial, robótica e sistemas embarcados, onde a precisão e eficiência do código gerado são cruciais. Sua capacidade de transformar modelos visuais em código funcional oferece aos engenheiros uma ferramenta poderosa para desenvolver, testar e implementar soluções em menor tempo e com maior confiabilidade (Lobo, 2017).

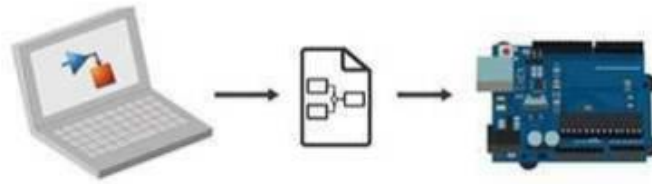
Além disso, sua compatibilidade com diversas plataformas e sua arquitetura extensível garantem que ele possa se adaptar às necessidades específicas de cada projeto, promovendo inovação e excelência no desenvolvimento de sistemas modernos.

#### **2.4.4 Pacotes de Suporte para Arduino**

O pacote de suporte do MATLAB e Simulink para Arduino é uma extensão poderosa que facilita o desenvolvimento de aplicações integradas entre software e hardware, utilizando a plataforma Arduino. Com ele, é possível criar códigos diretamente no MATLAB ou modelos no Simulink, que são automaticamente compilados, carregados e executados na placa Arduino, eliminando a necessidade de usar IDEs externas, como o Arduino IDE.

Esse pacote fornece uma interface direta para leitura e escrita nas entradas e saídas digitais e analógicas do Arduino, como pode-se observar na Figura 13, permitindo interagir facilmente com sensores, atuadores e outros dispositivos periféricos. Além disso, ele suporta comunicação com módulos adicionais, como displays, módulos de comunicação sem fio (Wi-Fi, Bluetooth) e sistemas de armazenamento de dados.

Figura 13 – Fluxograma de funcionamento do Simulink Support Package for Arduino



Fonte: (Lobo, 2017)

#### 2.4.4.1 Pacotes de Suporte do Simulink e Geração de Código

O pacote de suporte do Simulink permite carregar funcionalidades diretamente na plataforma, de modo que as funções só são executadas enquanto a placa estiver conectada. No entanto, ao utilizar a geração automática de código, o Arduino pode continuar executando o código mesmo quando desconectado, desde que seja alimentado por outra fonte de energia. Apesar dessa praticidade, os pacotes de suporte não oferecem compatibilidade com todas as placas nem suporte a todas as funcionalidades disponíveis na plataforma.

#### 2.4.4.2 Pacotes Adicionais e seus Benefícios

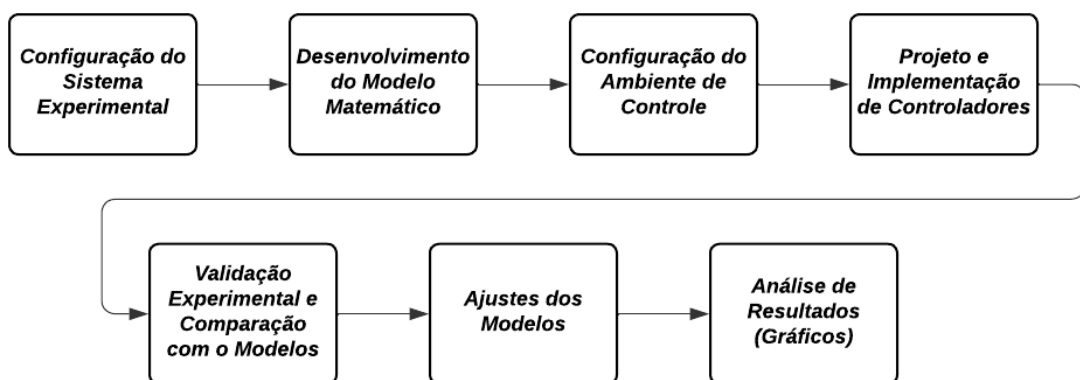
Para ampliar as possibilidades de uso, é possível instalar pacotes adicionais que disponibilizam blocos específicos para integração com o Arduino, incluindo sensores. Um exemplo é a Rensselaer Arduino Support Package Library (RASPLib), que permite a utilização de diversos sensores amplamente aplicados, como os que medem temperatura, distância, pressão e umidade, entre outras grandezas.

Os principais benefícios de usar os pacotes de suporte do MATLAB e Simulink são a facilidade para desenvolver algoritmos complexos e avançados, a possibilidade de modificá-los com agilidade e a redução de custos e tempo no desenvolvimento.

### 3 Materiais e Métodos

Neste capítulo, será apresentada a metodologia adotada para o desenvolvimento da aplicação de controle clássico na prática. As principais etapas do processo estão organizadas em tópicos e descritas a seguir. A Figura 14 ilustra de forma clara e objetiva o fluxo das etapas envolvidas na implementação da metodologia proposta.

Figura 14 – Fluxograma Geral



Fonte: do Autor

#### 3.1 Descrição do Sistema Experimental

O sistema experimental desenvolvido tem como objetivo a aplicação prática de conceitos de controle clássico, utilizando ferramentas acessíveis para integração entre software e hardware.

O Arduino sistema opera de forma integrada, onde o MATLAB e o Simulink são utilizados para aplicar e observar a resposta dinâmica do sistema, projetar controladores (como PID), e enviar os comandos para o Arduino via comunicação serial. O Arduino, por sua vez, controla o motor por meio da ponte H, ajustando a velocidade de acordo com o sinal de controle recebido. O encoder do motor fornece dados de feedback ao Simulink, permitindo a análise da resposta dinâmica do sistema e a comparação entre o comportamento modelado e real.

Essa configuração prática e interativa proporciona um ambiente de aprendizado dinâmico, onde ajustes podem ser feitos em tempo real, promovendo o entendimento dos conceitos teóricos aplicados ao controle clássico.

### 3.1.1 Componentes Utilizados

Os componentes utilizados neste projeto incluem, primeiramente, um motor DC, que será a nossa planta, frequentemente referida como  $G(s)$  nas aulas de controle. Este modelo é obtido a partir da modelagem matemática do sistema ou pela análise da resposta ao degrau experimentalmente. O motor DC representa o sistema físico a ser controlado, sendo essencial para validar os conceitos teóricos de controle.

Além disso, utilizamos uma ponte H, que converte o sinal PWM gerado pelo Arduino em um controle de tensão adequado para o motor. Essa ponte permite enviar sinais de controle com uma tensão maior, indispensável para o funcionamento do sistema, já que a tensão do Arduino é insuficiente para essa tarefa. Para alimentar o sistema, empregamos uma fonte de bancada, que fornece energia adequada ao motor e à ponte H. Por fim, uma protoboard é utilizada para interligar os componentes e realizar a montagem do circuito de forma prática e organizada, tendo o Arduino como o nosso controlador.

#### 3.1.1.1 Motor JGA25-370 DC6V280RPM

O motor da Figura 15 cujas características podem ser observadas nas Figuras 16 e 17, usa um conjunto de engrenagens para reduzir a velocidade e aumentar o torque. Isso torna o motor mais difícil de parar com uma força externa e permite carregar cargas maiores. Apesar de haver alguma perda de energia na desaceleração, ele oferece alta eficiência em aplicações que exigem grande torque, trazendo praticidade ao dia a dia (Seeed, 2024).

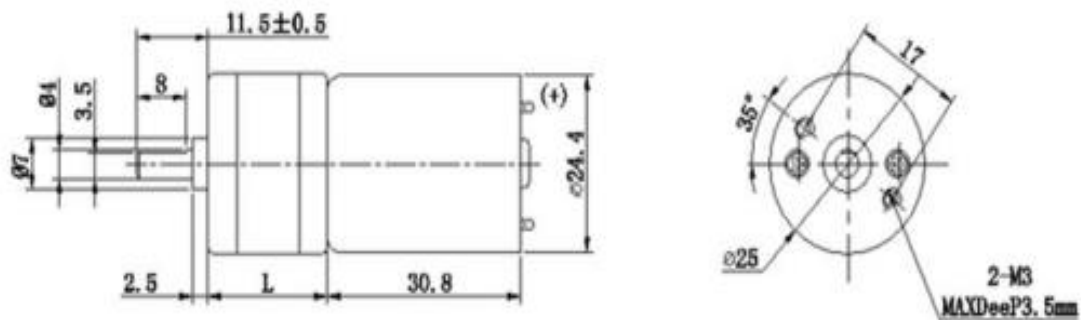
- Função: Planta de controle, cuja velocidade é ajustada e monitorada em tempo real.
- O encoder fornece feedback da velocidade angular, permitindo a implementação de estratégias de controle em malha fechada.

Figura 15 – Motor DC 6V 280 RPM com Roda e Encoder



Fonte: (Autocorerobotica, 2024)

Figura 16 – Motor DC 6V 280 RPM Dimensões



Fonte: (Seed, 2024)

Figura 17 – Motor DC 6V 280 RPM Especificações

Voltage V	No-load		Maximum efficiency pointed				Blockage	
	speed r/min	electric current A	speed r/min	electric current A	Torque Kg.cm	Power W	Torque Kg.cm	electric current A
6	280	0.2	133	0.5	0.75	1.1	4.0	2.1
12	350	0.1	245	0.65	1.4	2.4	5.2	2.2

Fonte: (Seed, 2024)

### 3.1.1.2 Ponte H L298N

A Ponte H que pode ser observada na Figura 18 será responsável por receber o sinal de entrada gerado pelo Simulink, transmitido para o sistema físico por meio da comunicação serial com o Arduino. O Arduino interpretará o sinal recebido do Simulink e ajustando os pinos de controle da Ponte H para determinar a direção e a amplitude da tensão aplicada ao motor. Dessa forma, ela controlará diretamente o motor, permitindo que o sistema físico responda às variações do sinal de controle.

Essa configuração cria uma integração e viabiliza a realização de testes e ajustes de controladores em tempo real. Além disso, permite validar a resposta dinâmica do motor em um ambiente controlado, aproximando teoria e prática de maneira prática e didática.

Figura 18 – Ponte H



Fonte: (Robocore, 2024)

### 3.1.1.3 Fonte de Bancada Yanxun OS-1502DD+

A fonte de bancada Yaxun 1502DD+ que pode ser observada na Figura 19 foi utilizada como fonte de alimentação principal para o motor DC e outros componentes do circuito. Esta fonte variável oferece uma tensão ajustável de até 15V e uma corrente máxima de 2A, características que a tornam ideal para experimentos em controle de sistemas. A capacidade de ajustar a tensão de saída de forma precisa permitiu simular diferentes condições de operação do motor, enquanto sua estabilidade garantiu a consistência dos testes, minimizando interferências externas.

Além disso, a Yaxun 1502DD+ oferece proteção contra sobrecarga e curto-circuito, aumentando a segurança dos experimentos e protegendo os componentes utilizados no projeto. O display digital integrado facilitou o monitoramento contínuo da tensão e da corrente durante as atividades, contribuindo para a confiabilidade das medições e ajustes. Essas características tornaram a fonte de bancada uma ferramenta indispensável para alimentar o sistema e assegurar a execução prática do controle proposto, alinhando a experimentação com os requisitos teóricos.

Figura 19 – Fonte de Bancada

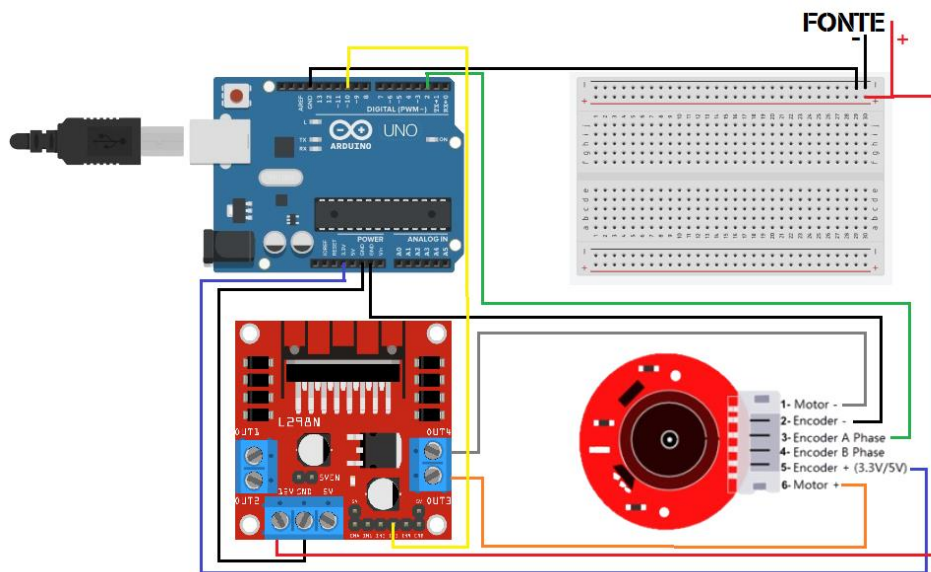


Fonte: (Riolink, 2024)

### 3.1.2 Montagem

A Figura 20 apresenta o diagrama de montagem do hardware utilizado neste projeto, que integra o microcontrolador Arduino UNO, um driver de motor L298N e um motor de corrente contínua (DC) com encoder. O **Arduino** é responsável por receber os sinais enviados pelo Simulink e transmiti-los ao sistema físico, além de retornar ao Simulink os dados de velocidade do motor, conforme especificado no código implementado. Esta configuração permite a criação de um sistema de controle dinâmico em malha fechada, com comunicação serial para troca de informações em tempo real.

Figura 20 – Diagrama de Montagem do Hardware do Sistema de Controle



Fonte: do Autor

A fonte de alimentação conecta-se ao driver L298N, fornecendo energia tanto para o motor quanto para o circuito eletrônico. O driver, por sua vez, controla o acionamento do motor, enquanto o Arduino gerencia o fluxo de dados e realiza a leitura dos sinais gerados pelo encoder. Este feedback fornecido pelo encoder é essencial para monitorar a posição e a velocidade do motor, possibilitando a análise da dinâmica do sistema e o ajuste preciso dos parâmetros de controle. Pode-se observar o hardware do sistema montado na Figura 21:

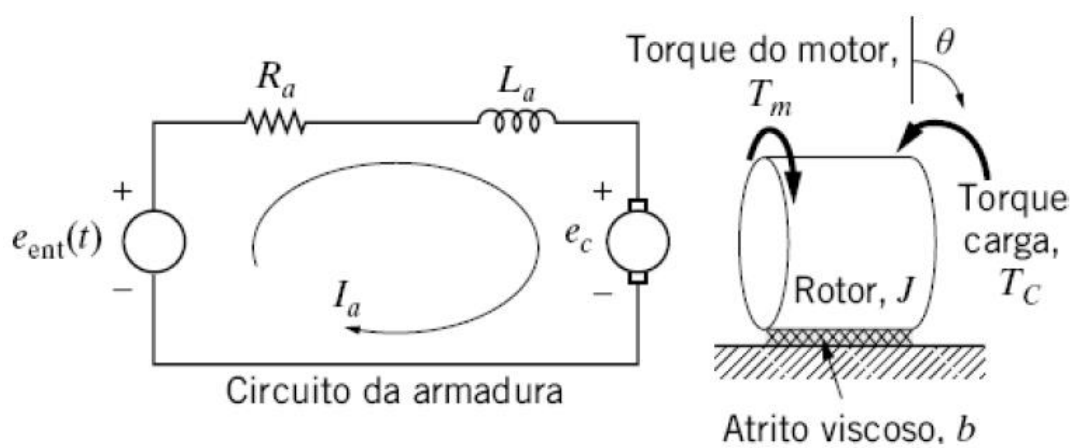
Figura 21 – Hardware do Sistema Montado



Fonte: do Autor

### 3.2 Desenvolvimento do Modelo Matemático

Figura 22 – Diagrama esquemático de um motor CC



Fonte: (Kluever, 2017)

A Figura 22 ilustra o diagrama utilizado para obter as equações diferenciais que descrevem a dinâmica do sistema a partir das leis físicas fundamentais. No caso de um motor de corrente contínua (CC), que é um sistema eletromecânico, aplicam-se as Leis de Kirchhoff para modelar a parte elétrica e as Leis de Newton para descrever a parte mecânica, analisando forças e torques.

Essas duas partes estão interligadas: o torque produzido pelo motor é proporcional à corrente elétrica, enquanto a força contra-eletromotriz, gerada pelo movimento do rotor, é proporcional à sua velocidade angular. Essa relação eletromecânica permite integrar as equações elétricas e mecânicas, proporcionando um modelo matemático completo do sistema, fundamental para o estudo de sua dinâmica e controle.

As variáveis e equações para a modelagem são:

$$T_m = Fr = B\ell r \quad (3.1)$$

Onde  $T_m$  é o torque eletromagnético total sobre o rotor,  $r$  é o raio do rotor e  $\ell$  é o comprimento total dos enrolamentos.

$$T_m = K_m I \quad (3.2)$$

Se a intensidade do campo magnético  $B$  é constante os três termos  $B\ell r$  podem ser concentrados em uma única constante,  $K_m = B\ell r$ , chamada de constante de torque do motor ( $Nm/A$ ).

A Equação 3.2 acima mostra como uma corrente aplicada em um motor de CC produz um torque fazendo-o girar.

O movimento angular do rotor produz uma tensão induzida contrária ao sentido da corrente na armadura (contra  $F_{em}$ ,  $e_c$ ).

$$e_c = B\ell v = B\ell r \dot{\theta} \quad (3.3)$$

Se o campo magnético é constante pode-se definir uma nova constante,  $K_c = B\ell r$ , chamada de constante de contra  $F_{em}$  ( $Vs/rad$ ). Logo:

$$e_c = K_c \dot{\theta} \quad (3.4)$$

A Equação 3.3 acima mostra como um deslocamento angular no rotor de um motor de CC produz uma tensão elétrica induzida.

A armadura possui várias bobinas que apresentam resistência  $R_a$  e indutância  $L_a$ . Assim, aplicando-se uma tensão de entrada  $e_{ent}$ :

$$e_{ent} - e_R - e_L - e_c = 0 \quad (3.5)$$

Portanto, juntando todas as equações, o modelo elétrico do motor será dado por:

$$L_a \dot{I}_a + R_a I_a + K_c \dot{\theta} = e_{ent} \quad (3.6)$$

O rotor é conectado a alguma massa (carga) que possui momento de inércia  $J$ , que no caso do trabalho é uma roda. Aplicando-se a segunda lei de Newton obtém-se (modelo mecânico):

$$K_m I_a - T_c - b \dot{\theta} = J_m \ddot{\theta} \quad (3.7)$$

Portanto, considerando que o torque da carga  $J_c$  e o torque do motor  $J_m$  são definidos pela multiplicação de seus respectivos momentos de inércia pela aceleração angular  $\ddot{\theta}$ , e sabendo que ambos compartilham o mesmo eixo, suas acelerações angulares são iguais. Dessa forma, é possível combinar  $(J_c + J_m)$  os dois termos em uma única expressão, representando o momento de inércia total do sistema  $J_{total}$ , que é a soma do momento de inércia do motor com o da carga, multiplicado pela aceleração angular. Dessa forma reduzimos a Equação 3.7 para a Equação 3.8:

$$K_m I_a - b \dot{\theta} = J_{total} \ddot{\theta} \quad (3.8)$$

Dessa forma, obtém-se a modelagem matemática completa do motor que defini a sua dinâmica, por meio das seguintes equações diferenciais:

$$\begin{cases} L_a \dot{I}_a + R_a I_a + K_c \dot{\theta} = e_{ent} \\ K_m I_a - b \dot{\theta} = J_{total} \ddot{\theta} \end{cases} \quad (3.9)$$

$$w = \dot{\theta} \quad (3.10)$$

$$\dot{w} = \ddot{\theta} \quad (3.11)$$

$$\begin{cases} L_a \dot{I}_a + R_a I_a + K_c w = e_{ent} \\ K_m I_a - b w = J_{total} \dot{w} \end{cases} \quad (3.12)$$

- $w$  é a velocidade angular do motor.

### 3.2.1 Função de Transferência do Motor

Para obter a função de transferência do motor, é necessário converter o modelo do domínio do tempo para o domínio da frequência. Essa conversão é realizada por meio da aplicação da transformada de Laplace às equações diferenciais que descrevem o sistema. Esse processo facilita a análise do comportamento dinâmico do motor no contexto de sistemas

lineares e invariantes no tempo, permitindo trabalhar com funções algébricas no lugar de equações diferenciais.

Além disso, é fundamental definir as variáveis de entrada e saída do sistema. No caso em análise, a entrada do sistema é representada pela tensão de alimentação  $E_{ent}(s)$ , enquanto a saída corresponde à velocidade angular do motor  $W(s)$ . A função de transferência é obtida como a relação entre a transformada de Laplace da saída e a da entrada. Assim, ao aplicar uma determinada tensão de entrada, o motor responde gerando movimento, caracterizado por sua velocidade angular.

Aplicando-se a transformada de Laplace na Equação 3.12 obtém-se:

$$\frac{W(s)}{E_{ent}(s)} = \frac{K_m}{LJ_{total}s^2 + (RJ_{total} + LB)s + RB + K_mK_c} \quad (3.13)$$

### 3.2.2 Parâmetros do motor estabelecidos através de ensaios

- 1° Passo: Medir a resistência da armadura nos terminais do Motor DC. Desconecte o motor do resto do circuito e coloque as pontas de prova do multímetro nos terminais do motor, lembrando de ajustar o multímetro para medição de resistência.

$$R_a \cong 3,7 \Omega \quad (3.14)$$

- 2° Passo: Medir a indutância da armadura nos terminais do Motor DC. Desconecte o motor do resto do circuito e coloque as pontas de prova do multímetro nos terminais do motor, lembrando de ajustar o multímetro para medição de indutância,

$$L_a \cong 1,5mH \quad (3.15)$$

Obs: Devido a ausência de um indutímetro para a medição direta, a indutância do motor foi estimada com base em valores típicos. A escolha de 1,5 mH como indutância para o motor DC é justificada por estar dentro da faixa típica de 1 a 10 mH encontrada em motores pequenos. Esse valor reflete o tamanho reduzido dos enrolamentos e a construção compacta do rotor, características comuns em motores do tipo. A estimativa de 1,5 mH é uma aproximação razoável na ausência de medições precisas ou dados do fabricante.

- 3° Passo: Achar  $K_c$  (constante contra-fem). Aplicar uma tensão nos terminais de entrada do motor ( $e_{ent}$ ) e observar a corrente ( $I$ ) e velocidade ( $w$ ).

$$e_c = K_c \dot{\theta} = K_c w \quad (3.16)$$

$$e_c = e_{ent} - R_a \cdot I \quad (3.17)$$

$$K_c \cdot w = e_{ent} - R_a \cdot I \quad (3.18)$$

$$K_c = \frac{e_{ent} - R_a \cdot I}{w} \quad (3.19)$$

$$K_c = \frac{(3V) - (3,7\Omega) \cdot (96,5 \cdot 10^{-3}A)}{\left(\frac{16,83 \text{ rad}}{s}\right)} \quad (3.20)$$

$$K_c \cong 0,157 \text{ Vs/rad} \quad (3.21)$$

$$K_m \cong 0,157 \text{ N} \cdot \text{m/A} \quad (3.22)$$

Obs: Para motores DC simples, as constantes  $K_c$  e  $K_m$  são iguais. Foi realizado apenas um ensaio; entretanto, o ideal seria realizar múltiplos ensaios aplicando tensões entre 0 e 6 V, calculando a média dos valores de  $K_c/K_m$  obtidos. Posteriormente, observou-se que, ao substituir esse valor na função de transferência, o sistema foi representado de maneira adequada. Portanto, para esse caso específico, um ensaio foi suficiente.

- 4º Passo: Achar ( $\tau$ ) constante de tempo. Tempo que demora para o sistema chegar em 63,2% do valor final.

Obs: Para determinar a constante de tempo, é essencial visualizar o gráfico correspondente. Essa etapa será realizada após a comunicação serial entre o Simulink e o Arduino ser devidamente estabelecida, permitindo a aquisição e o monitoramento dos dados necessários.

- 5º Passo: Achar momento de inércia ( $J_{total}$ ).

Obs: A fórmula para encontrarmos esse parâmetro depende da constante de tempo, portanto, será definido posteriormente.

- 6º Passo: Achar constante de fricção de coloumb ( $B$ ).

$$B = \frac{T_m - T_f}{w(t)} \quad (3.23)$$

$$B = \frac{I(t) \cdot K_m - T_f}{w(t)} \quad (3.24)$$

$$B = \frac{(96,5mA) \cdot \left(\frac{0,157 \text{ Nm}}{A}\right) - 0}{\left(\frac{16,83 \text{ rad}}{s}\right)} \quad (3.25)$$

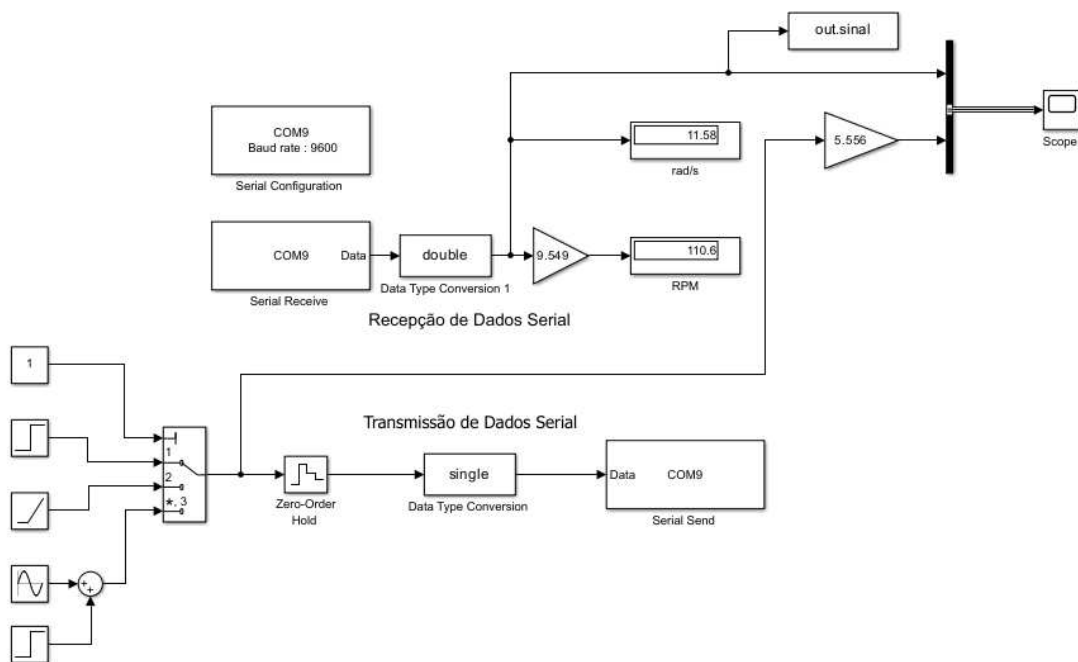
$$B = 0,000900207962 \text{ N} \cdot \text{m} \cdot \text{s} \quad (3.26)$$

Obs:  $T_f$  é o torque de fricção, para esse caso foi considerado igual a zero pois não temos uma carga relativamente alta.

### 3.3 Comunicação Serial: Arduino e Simulink

A Figura 23 apresenta um modelo de comunicação serial bidirecional entre o Arduino e o Simulink. No bloco denominado "Recepção de Dados Serial", o sistema recebe informações do Arduino via porta serial configurada no Simulink.

Figura 23 – Diagrama Iterativo no Simulink em Malha Aberta



Fonte: do Autor

Os dados recebidos, inicialmente no formato bruto, são convertidos para o tipo de dado apropriado (neste caso, "double") e escalados para representar grandezas físicas, como velocidade em rad/s e RPM. Esses valores são visualizados em monitores de bloco para fins de análise do sistema em malha aberta e podem ser utilizados em subsistemas adicionais para controle ou monitoramento.

No lado oposto, o bloco "Transmissão de Dados Serial" é responsável por enviar comandos ou valores do Simulink para o Arduino. A entrada do sistema de transmissão recebe um sinal gerado por diferentes fontes (como ondas senoidais, degrau, ou um somatório desses), permitindo testes e análises do comportamento do motor ou do sistema em estudo. Antes de ser

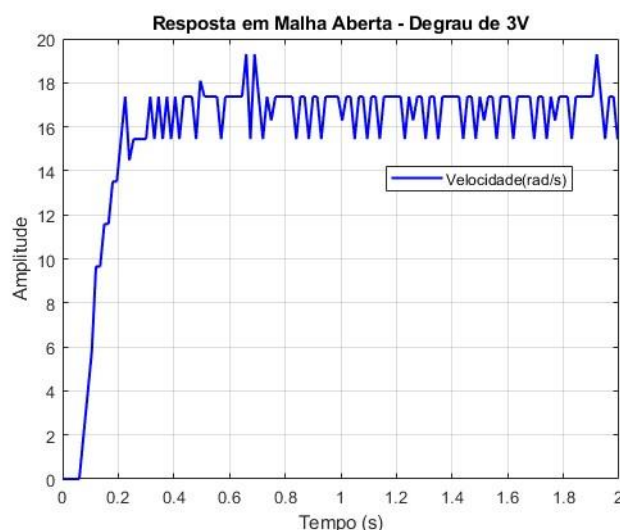
enviado, o sinal é convertido para o tipo "single" para compatibilidade com a transmissão serial, garantindo a correta interpretação dos dados pelo Arduino.

Esse esquema é essencial para a integração entre hardware e software no contexto acadêmico, permitindo que o Simulink sirva como uma interface de modelagem e controle para o Arduino. Ele exemplifica como é possível validar algoritmos de controle no ambiente Simulink e implementá-los diretamente em hardware acessível, promovendo a experimentação prática e a aplicação dos conceitos teóricos, algo fundamental em projetos de ensino de engenharia e controle.

O tempo de amostragem de 25 ms foi inicialmente escolhido com base no critério de Nyquist, que estabelece que a frequência de amostragem deve ser pelo menos duas vezes maior que a frequência máxima do sistema, relacionada à frequência de corte. Essa frequência de corte foi determinada posteriormente, ao encontrar a função de transferência completa que representa o sistema, seja pelo modelo matemático ou experimental. A partir da análise do diagrama de Bode, verificou-se que a frequência de corte era aproximadamente 9 rad/s; multiplicando por dois, obteve-se 18 rad/s, e ao inverter esse valor, encontrou-se um tempo de amostragem de cerca de 55,5 ms, suficiente para capturar a dinâmica do sistema. Optou-se, então, por um valor menor, e após testes, 25 ms mostrou-se o mais adequado para o caso.

A partir do gráfico da Figura 24, onde a amplitude é a **velocidade** em rad/s, é possível determinar a constante de tempo do sistema, uma grandeza fundamental na análise de sistemas dinâmicos. A constante de tempo é obtida observando o tempo necessário para que a resposta do sistema alcance aproximadamente 63,2% do seu valor final em um comportamento de primeira ordem.

Figura 24 – Curva de Reação



Fonte: do Autor

Essa análise permite avaliar a velocidade de resposta do sistema e sua capacidade de adaptação a mudanças na entrada. No caso analisado, a constante de tempo do sistema foi calculada e resultou na Equação 3.27, refletindo a dinâmica específica do sistema sob as condições estudadas.

$$\tau = 0,083s \quad (3.27)$$

Com essa informação, agora pode-se encontrar o valor do momento de inércia:

$$J = \frac{0,083s \cdot \frac{0,157Vs}{rad} \cdot \frac{0,157Nm}{rad}}{3,7\Omega} \quad (3.28)$$

$$J = 0,000552937027Kg \cdot m^2 \quad (3.29)$$

Portanto, agora temos a lista completa dos parâmetros dada por:

Tabela 3 – Lista completa dos parâmetros

Parâmetros	Valores
Momento de Inércia (J)	0,000552937027Kg · m <sup>2</sup>
Constante de Fricção (B)	0,000900207962 N · m · s
Const. F. Eletromotriz (Kc)	0.157 Vs/rad
Constante Mecânica (Km)	0.157 N · m/A
Resistência da Armadura (R)	3.7 Ω
Indutância da Armadura (L)	0.0015H

Fonte: do Autor

Substituindo os parâmetros na Equação 3.13, obtém-se o modelo matemático do motor:

$$\frac{W(s)}{E_{ent}(s)} = \frac{0,157}{0,0000082940554 \cdot s^2 + 0,00204721731s + 0,0279797695} \quad (3.30)$$

### 3.4 Definindo Função de Transferência Experimentalmente

A definição da função de transferência experimental por meio da curva de reação é uma abordagem prática para modelar sistemas de controle. Baseia-se na análise da resposta temporal de um sistema a um estímulo conhecido, como um degrau unitário, para identificar parâmetros como ganho estático, constante de tempo e atraso. Esses dados permitem derivar um modelo matemático que representa o comportamento dinâmico do sistema, sendo útil especialmente quando suas características não são conhecidas analiticamente.

Essa metodologia conecta teoria e prática, ajudando alunos a compreenderem como sistemas reais podem ser representados por modelos simplificados. A análise da curva de reação também revela efeitos como atrito e atrasos, destacando as limitações dos modelos. Dessa forma, a técnica torna-se uma ferramenta essencial para ensinar controle clássico com base em experimentos reais.

$$\frac{C(s)}{U(s)} = \frac{K}{Ts + 1} e^{-Ls} \quad (3.31)$$

Para que seja calculado o ganho K, serão usados os dados obtidos na Figura 24.

$$K = \frac{\text{Valor}_{final}}{\text{Entrada}} \quad (3.32)$$

$$K = \frac{\frac{16.83rad}{s}}{3V} \quad (3.33)$$

$$K = 5.543Vs/rad \quad (3.34)$$

O valor da constante de tempo encontra-se na Equação 3.27. O atraso de tempo do sistema foi definido a partir da Figura 24, que resultou aproximadamente em **0.075s**.

$$\frac{C(s)}{U(s)} = \frac{5.543}{0.083s + 1} e^{-0.075s} \quad (3.35)$$

Portanto, a Equação 3.35 apresenta a função de transferência de primeira ordem com atraso de tempo do motor, determinada de forma experimental. Essa equação também pode ser encontrada com mais detalhes no referencial teórico do trabalho, onde é apresentada como a Equação 2.14.

### 3.5 Parâmetros dos Controladores

Utilizando a regra de sintonia de Ziegler-Nichols para o método da curva de reação, representada na Tabela 1 do referencial teórico deste trabalho, é possível determinar os parâmetros dos controladores Proporcional (P), Proporcional-Integral (PI) e Proporcional-Integral-Derivativo (PID). Esse método baseia-se na análise da resposta do sistema a um estímulo de entrada, permitindo identificar características como ganho estático, constante de tempo e atraso.

A partir desses dados, as tabelas propostas por Ziegler-Nichols fornecem ajustes específicos para cada tipo de controlador, otimizando seu desempenho em termos de estabilidade, tempo de resposta e rejeição a perturbações. Dessa forma, a técnica proporciona uma abordagem prática e sistemática para projetar controladores eficientes.

- Controlador Proporcional:

$$P = K_p \quad (3.36)$$

$$K_p = \frac{T}{L} \quad (3.37)$$

$$K_p = \frac{0.083}{0.075} \quad (3.38)$$

$$P = 1.107 \quad (3.39)$$

- Controlador Integral:

$$PI = K_p + \frac{K_i}{s} \quad (3.40)$$

$$K_p = 0.9 \cdot \frac{T}{L} \quad (3.41)$$

$$K_p = 0.9 \cdot \frac{0.083}{0.075} \quad (3.42)$$

$$K_p = 0.996 \quad (3.43)$$

$$T_i = \frac{L}{0.3} \quad (3.44)$$

$$T_i = \frac{0.075}{0.3} \quad (3.45)$$

$$T_i = 0.25 \quad (3.46)$$

$$K_i = \frac{K_p}{T_i} \quad (3.47)$$

$$K_i = \frac{0.996}{0.25} \quad (3.48)$$

$$K_i = 3.984 \quad (3.49)$$

$$PI = 0.996 + \frac{3.984}{s} \quad (3.50)$$

- Controlador Proporcional Integral Derivativo:

$$PID = K_p + \frac{K_i}{s} + K_d s \quad (3.51)$$

$$K_p = 1.2 \frac{T}{L} \quad (3.52)$$

$$K_p = 1.2 \frac{0.083}{0.075} \quad (3.53)$$

$$K_p = 1.328 \quad (3.54)$$

$$T_i = 2L \quad (3.55)$$

$$T_i = 2 \cdot 0.075 \quad (3.56)$$

$$T_i = 0.15 \quad (3.57)$$

$$T_d = 0.5L \quad (3.58)$$

$$T_d = 0.5 \cdot 0.075 \quad (3.59)$$

$$T_d = 0.0375 \quad (3.60)$$

$$K_i = \frac{K_p}{T_i} \quad (3.61)$$

$$K_i = \frac{1.328}{0.15} \quad (3.62)$$

$$K_i = 8.853 \quad (3.63)$$

$$K_d = K_p T_d \quad (3.64)$$

$$K_d = 1.328 \cdot 0.0375 \quad (3.65)$$

$$K_d = 0.0498 \quad (3.66)$$

$$PID = 1.328 + \frac{8.853}{s} + 0.0498s \quad (3.67)$$

### 3.6 Ajustes e Complementos no Sistema

Como pode ser observado na Figura 24, o gráfico apresenta uma grande quantidade de ruídos, que são decorrentes das características físicas do sistema, das medições realizadas e também da influência da simulação. Diante dessas interferências, é necessário suavizar o sinal para que a resposta do sistema seja apresentada de maneira mais clara e precisa, facilitando a análise pelos alunos. Para isso, será adicionado um filtro passa-baixa à saída, que ajudará a eliminar essas flutuações indesejadas, proporcionando uma visualização mais adequada do comportamento do sistema.

Além disso, a função de transferência obtida inicialmente na modelagem matemática não leva em consideração o atraso de tempo presente no sistema. Para corrigir isso, será utilizado o método de aproximação de Padé, disponível no MATLAB. A aproximação de Padé é uma técnica que visa aproximar funções com atraso de tempo por meio de uma representação racional, utilizando séries de Taylor para criar um modelo de ordem finita que represente o comportamento do sistema com o atraso.

Nesse caso, será aplicada uma aproximação de segunda ordem, que equilibra a precisão com a complexidade, evitando tornar o modelo excessivamente complicado para o projeto dos controladores pelos alunos. Por fim, será desenvolvido um diagrama de malha fechada no Simulink para permitir a visualização do funcionamento dos controladores, visto que até o momento o sistema foi modelado em malha aberta. Os gráficos também poderão ser visualizados no MATLAB, utilizando o bloco "Out" do Simulink para realizar a análise dos resultados.

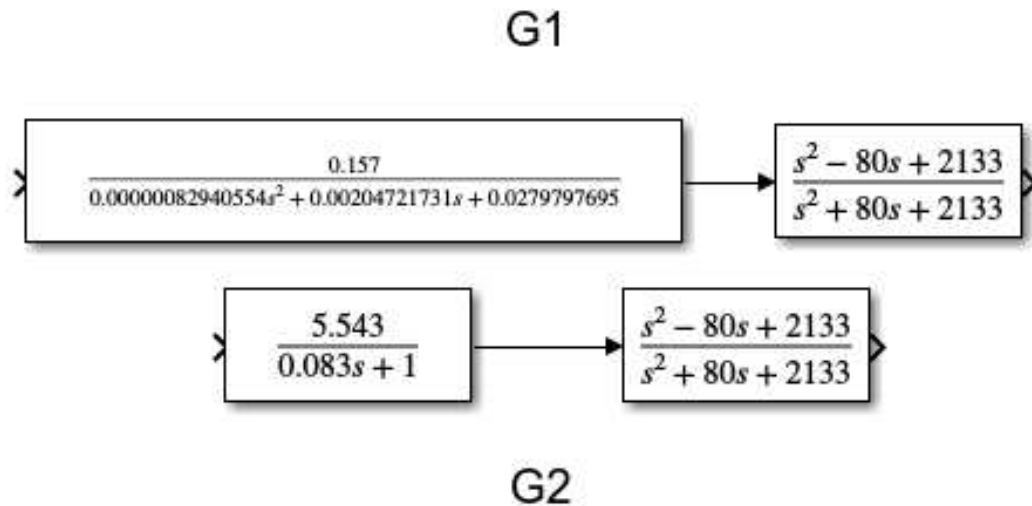
#### 3.6.1 Funções de Transferência com atraso através da aproximação de padé

Através dos cálculos e experimentos realizados, foram obtidos dois modelos para o sistema, os quais serão denominados como planta G1 e planta G2 das Equações 3.30 e 3.35. Para levar em consideração os efeitos de atraso de tempo na função de transferência obtida matematicamente, e uma vez que a função experimental já incorpora esse atraso, será utilizada a aproximação de Padé. Com isso, será formada uma nova função de transferência que

representará o sistema de maneira mais precisa, incluindo o atraso de tempo, permitindo uma modelagem mais realista para o controle e análise do sistema.

Adicionando-se o atraso de tempo na forma de função de transferência, obtém-se os modelos que podem ser observados na Figura 25:

Figura 25 – Funções de Transferência com Atraso



Fonte: do Autor

### 3.6.2 Filtro passa-baixa para suavização do sinal

Para implementar o filtro passa-baixa, é fundamental determinar a frequência de corte do sistema, que define a fronteira entre as frequências que serão atenuadas e aquelas que serão preservadas. Para calcular essa frequência, será necessário gerar o diagrama de Bode da função de transferência do sistema, que representa a magnitude e a fase do sistema em função da frequência. Esse diagrama pode ser obtido utilizando ferramentas como o MATLAB ou o Simulink, a partir da função de transferência que foi modelada matematicamente. No gráfico de Bode, a frequência de corte é identificada como o ponto em que a magnitude da resposta do sistema cai 3 dB abaixo do valor máximo ou de referência.

Após identificar a frequência de corte, a escolha da frequência de operação do filtro passa-baixa deve ser feita de forma cuidadosa. A frequência do filtro será definida ligeiramente acima da frequência de corte, de modo a garantir que a dinâmica importante do sistema, que ocorre em frequências mais baixas, seja preservada. O filtro precisa atenuar as frequências indesejadas, como ruídos e interferências de alta frequência, sem comprometer a resposta do sistema nas frequências que são relevantes para o seu funcionamento.

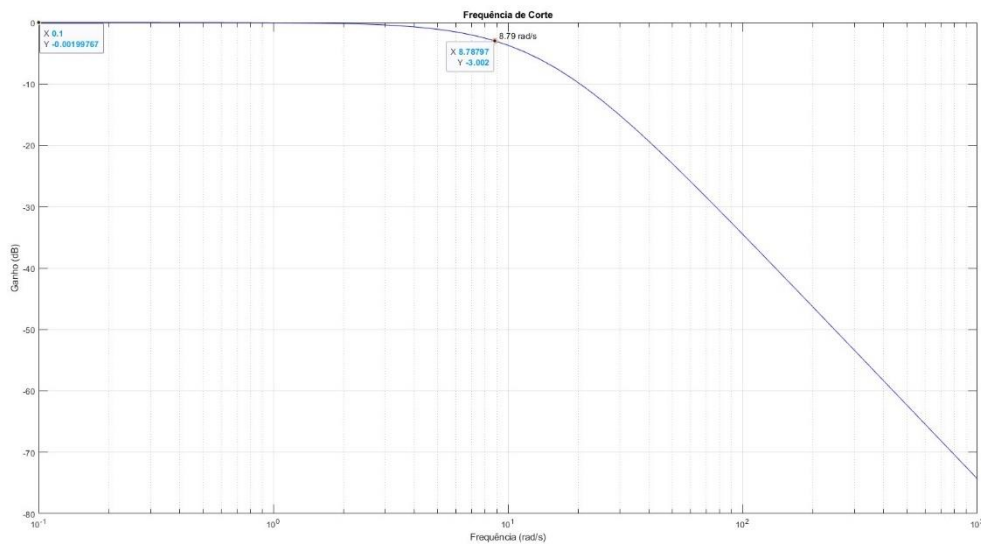
A escolha apropriada da frequência de corte e da frequência do filtro tem uma relação direta com a eficácia do processo de filtragem e com a qualidade da resposta do sistema. Ao definir uma frequência de filtro ligeiramente acima da de corte, busca-se criar um equilíbrio entre a remoção de componentes indesejáveis e a preservação das características dinâmicas do sistema. A análise cuidadosa desses parâmetros permite obter um filtro que otimize a performance do sistema, facilitando a observação e o entendimento da sua resposta pelos alunos.

O filtro é dado pela seguinte função de transferência:

$$f_{\text{filtro}} = \frac{\omega_c}{s + \omega_c} \quad (3.68)$$

A partir dessa função de transferência, é possível implementar o filtro diretamente no Simulink, aproveitando sua interface gráfica intuitiva e a facilidade para simular sistemas dinâmicos. O uso da função de transferência no ambiente do Simulink permite configurar o filtro de forma precisa, ajustando parâmetros como ganhos, polos e zeros para atender às especificações do sistema.

Figura 26 – Obtendo frequência de corte do sistema



Fonte: do Autor

Considerando que a frequência de corte do sistema é 8,79 rad/s, como pode-se observar na Figura 26, a escolha da frequência do filtro será feita ligeiramente acima desse valor. Assim, optou-se por 16 rad/s, garantindo que o filtro atue de forma eficaz, preservando as dinâmicas

relevantes do sistema e atenuando adequadamente as frequências indesejadas, o ideal seria escolher uma frequência três vezes maior que a de corte, no entanto, após alguns testes percebeu-se que esse valor era razoável. No receio de que esse filtro influenciasse na dinâmica do sistema mais do que o desejado, ele também foi adicionado na função de transferência que representa o sistema.

Portanto, a função de transferência do filtro resulta em:

$$f_{\text{filtro}} = \frac{16}{s + 16} \quad (3.69)$$

### 3.6.3 Diagrama em Malha Fechada no Simulink e Gráficos via Matlab

Para construir o diagrama em malha fechada, basta seguir o mesmo princípio utilizado no diagrama em malha aberta. No entanto, em vez de aplicar diretamente o sinal de entrada ao sistema, o sinal de controle será colocado na entrada, ajustando a resposta do sistema de acordo com o controlador. Essa abordagem pode ser observada na Figura 27, onde está sendo feita a comparação da saída do sistema real com o modelo matemático e o modelo experimental.

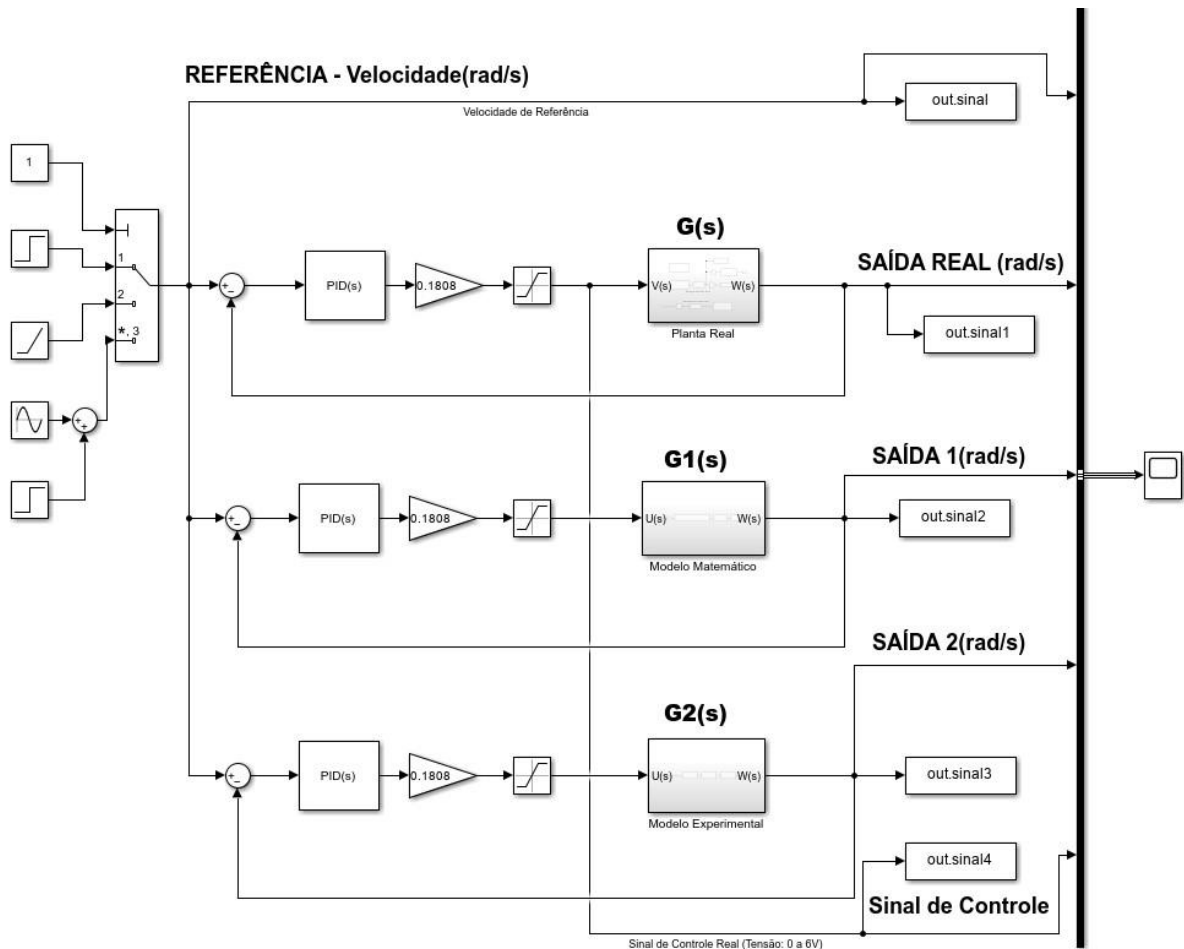
A Figura 27 apresenta o diagrama de um sistema de controle implementado no Simulink, utilizado para validar modelos matemáticos e experimentais de um motor DC em comparação com sua resposta real. O sistema recebe uma referência de entrada, que pode ser configurada de diferentes formas, como degrau, rampa e senoide. Essa entrada (velocidade) é enviada ao motor através de uma comunicação serial, permitindo que o sistema controle e registre a velocidade angular real (em rad/s). Os dados recebidos pelo sistema são processados para exibir a saída real do motor, que será utilizada na análise comparativa com os modelos teóricos.

Os modelos teóricos, representados por funções de transferência descrevem matematicamente o comportamento do motor DC. Esses modelos incluem os parâmetros característicos do motor, como inércia e constante de torque, e são usados para prever a resposta dinâmica do sistema. A saída dos modelos é comparada graficamente com a saída real medida, possibilitando avaliar a precisão e a fidelidade dos modelos em relação ao comportamento físico do motor. A primeira função de transferência foi obtida modelando o sistema matematicamente, e a segunda foi encontrada experimentalmente.

Esse sistema integra ferramentas de software e hardware, conectando o Simulink ao motor DC por meio de comunicação serial. Ele ilustra a aplicação prática do controle clássico, permitindo que modelos teóricos sejam validados experimentalmente. Essa abordagem une teoria e prática, sendo uma ferramenta valiosa no ensino de engenharia de controle, ao

demonstrar como sistemas reais respondem a diferentes estratégias de controle e modelagem matemática. Os dados podem ser observados **Matlab** através do bloco simout do simulink.

Figura 27 – Diagrama em Malha Fechada



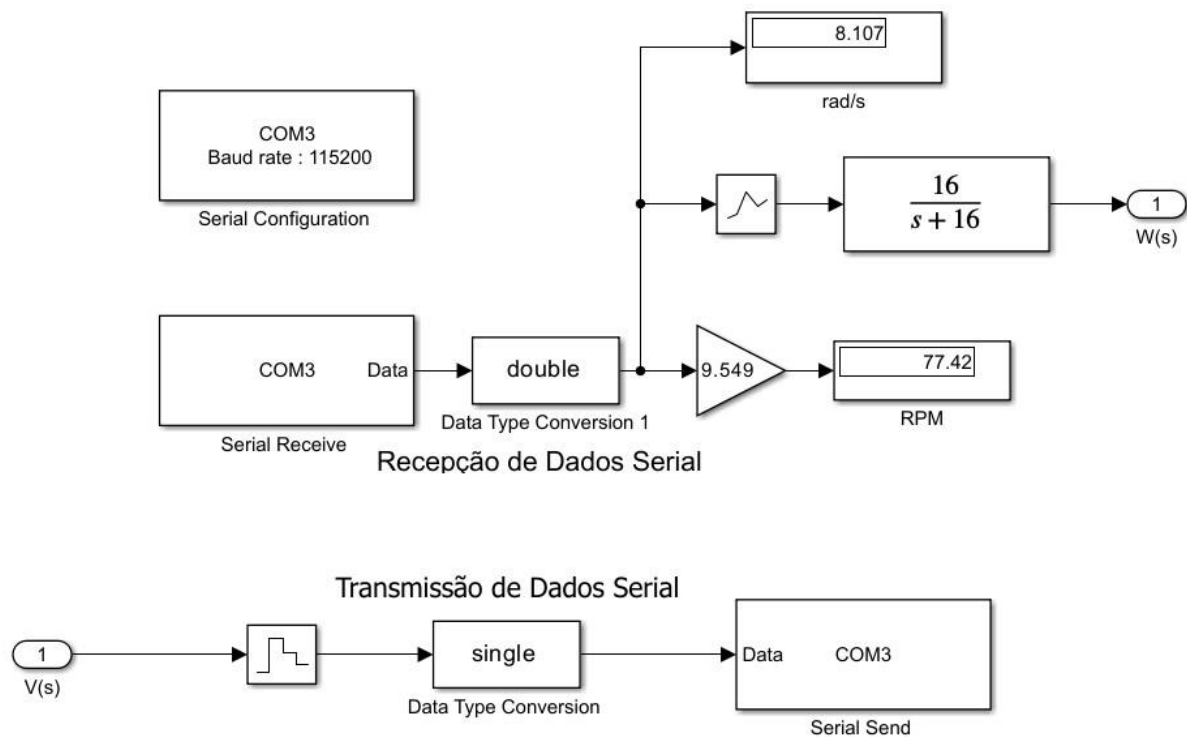
Fonte: do Autor

Como mostrado na Figura 28, o subsistema representa a planta real. O sinal de controle enviado à planta é uma tensão no intervalo de 0 a 6V, responsável por acionar o movimento do motor. Esse valor é transmitido via serial para o sistema, enquanto a velocidade do motor é recebida pelo Simulink, permitindo a implementação do feedback. No sistema em malha fechada, a entrada é um valor de velocidade.

Haja vista que a entrada do controlador corresponde ao erro de velocidade, olhando para a Figura 27, sua saída seria um sinal de velocidade. No entanto, para controlar o motor, é necessário enviar uma tensão. O ganho de 0,1808 realiza a conversão do sinal de velocidade para tensão, considerando a relação proporcional entre essas grandezas, que é determinada pelo ganho do sistema.

Como ilustrado na Figura 28, o valor de velocidade recebido pelo Simulink é discreto, sendo atualizado conforme o tempo de amostragem definido no código do Arduino, onde o cálculo da velocidade é realizado em intervalos específicos. Como utilizamos um filtro contínuo, é necessário processar esse valor antes de enviá-lo ao filtro. Para isso, o sinal passa pelo bloco 'First Order', que realiza a reconstrução do sinal no tempo contínuo por meio de interpolação.

Figura 28 – Subsistema / Planta Real



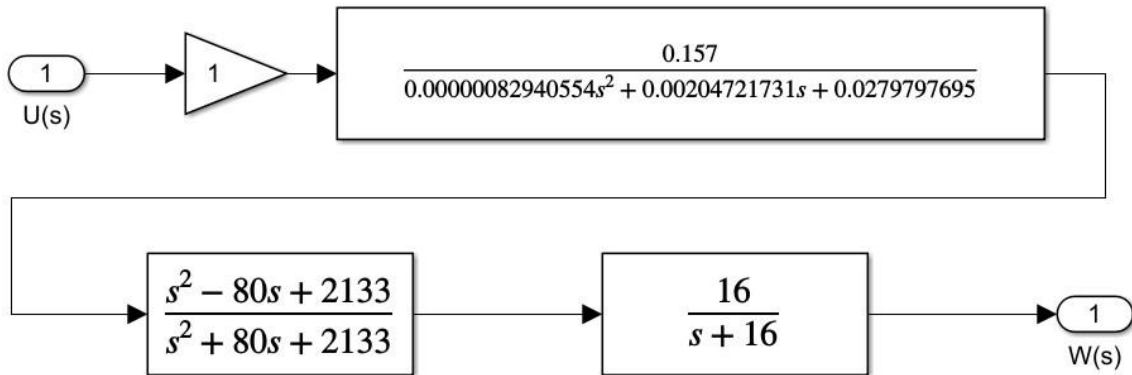
Fonte: do Autor

Os subsistemas apresentados nas Figuras 29 e 30 ilustram, respectivamente, os modelos matemático e experimental da nossa planta. Esses modelos consideram tanto o atraso de tempo quanto o filtro utilizado, permitindo uma representação mais precisa e alinhada com as características reais do sistema. Dessa maneira, conseguimos abordar o comportamento do sistema de forma mais abrangente, incorporando os efeitos dinâmicos relevantes.

Caso seja necessário realizar algum ajuste fino no modelo matemático, foi incluída a possibilidade de corrigir o ganho por meio do bloco *Gain* do Simulink. Para este sistema em particular, o modelo matemático demonstrou uma excelente correspondência com o ganho do sistema real, conforme verificado durante os testes realizados. Essa abordagem garante

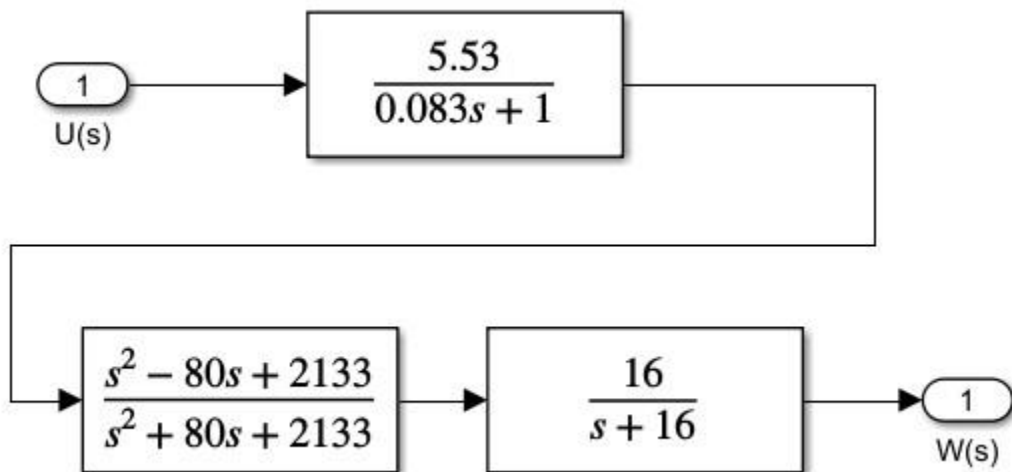
flexibilidade para ajustes adicionais, caso sejam identificadas discrepâncias em situações específicas.

Figura 29 – Subsistema / Modelo Matemático



Fonte: do Autor

Figura 30 – Subsistema / Modelo Experimental



Fonte: do Autor

### 3.6.4 Relação não linear entre o sinal PWM e a tensão enviada para o motor

Ao realizar experimentos em malha aberta, foi identificado que a relação entre o sinal de entrada (tensão de referência enviada pelo Simulink) e a tensão que efetivamente chega ao motor não era linear. Em tensões mais baixas, a tensão medida na saída da ponte H era maior do que a tensão configurada no degrau do Simulink. À medida que o valor da tensão de referência aumentava, essa diferença diminuía progressivamente até que, em um determinado

ponto, as tensões de entrada e saída se igualaram, como pode-se observar na Tabela 4, a consequência disso na velocidade do motor poder ser vista na Figura 32. Essa não linearidade pode ser explicada por características intrínsecas da ponte H e do sinal PWM, como as perdas internas nos componentes, as limitações do *duty cycle*, e o comportamento não ideal de comutação. Isso evidenciou a necessidade de um ajuste para garantir a precisão do controle e a fidelidade do modelo.

Para solucionar esse problema, foi realizada uma regressão linear e também a polinomial entre os pontos medidos, o que permitiu mapear com maior precisão a relação entre o valor configurado no Simulink e a tensão efetivamente aplicada ao motor. Esse ajuste não apenas corrigiu a diferença observada, mas também fez com que o nosso sistema fosse melhor representado através de polinômios de ordem superior, como ilustrado nas Figuras 33 e 34. A regressão desempenhou um papel crucial nesse processo, pois possibilitou a calibração do sistema e a obtenção de um modelo mais consistente, alinhando teoria e prática.

Tabela 4 – Resumo de dados para Regressão

Tensão de Entrada(V)	Tensão de Saída(V)
6	6
5	5.2
4	4.4
3	3.4
2	2.6
1	1.6

Fonte: do Autor

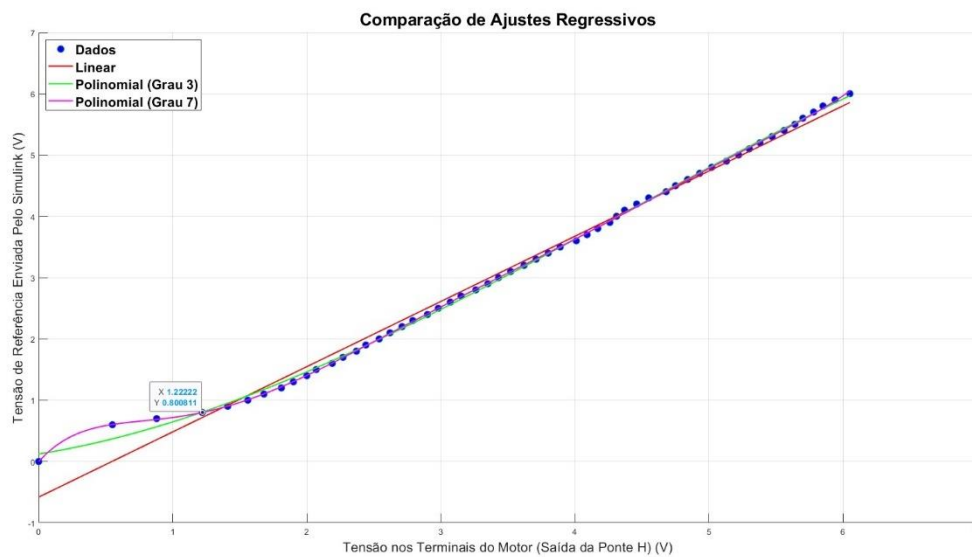
$$y = 0,001x^7 - 0,0237x^6 + 0,2274x^5 - 1.1202x^4 + \dots \\ \dots 2.9452x^3 - 3,7457x^2 + 2,4361x + 0.0003 \quad (3.70)$$

O polinômio que melhor representou os pontos apresentados na Figura 31 foi o de sétima ordem, conforme descrito na Equação 3.70. Para determinar esse polinômio, foram aplicadas tensões incrementais de 0,1V, registrando-se os pontos resultantes. Essas entradas foram

configuradas e geradas no Simulink, enquanto as saídas correspondentes foram medidas diretamente na saída da ponte H. Vale destacar que o motor estava conectado ao sistema durante a coleta desses dados, garantindo que as medições considerassem todas as características reais do sistema, incluindo dinâmicas e não linearidades.

Essa abordagem foi fundamental para realizar a pré-compensação dos dados, ajustando as condições do sistema para refletir seu comportamento real. No polinômio identificado,  $y$  representa o valor pré-compensado, enquanto  $x$  corresponde à tensão aplicada pelo Simulink. No contexto do sistema em malha aberta,  $x$  é diretamente a tensão de entrada. Já no sistema em malha fechada,  $x$  representa o sinal de controle, que também é expresso em forma de tensão.

Figura 31 – Ajustes Regressivos



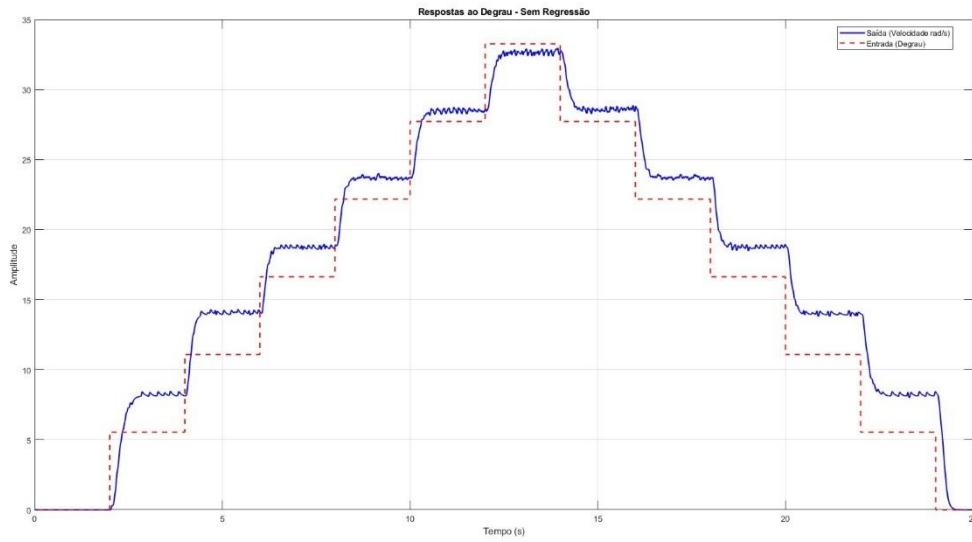
Fonte: do Autor

Portanto, tanto no sistema em malha aberta quanto no de malha fechada, o sistema nunca receberá exatamente o valor enviado diretamente pelo Simulink. Em vez disso, ele processará o valor ajustado pelo polinômio que descreve a curva de sétima ordem. Essa abordagem permite incorporar diversas características reais do sistema, o que possibilita atender às especificações da resposta transitória na maior parte das faixas de operação, algo incomum no controle clássico. No entanto, em velocidades muito baixas, o comportamento ideal pode não ser plenamente alcançado.

Isso ocorre porque o atrito crescente nessa faixa altera a constante de tempo do sistema, modificando o modelo real. Apesar dessa limitação, a pré-compensação de dados contribui para

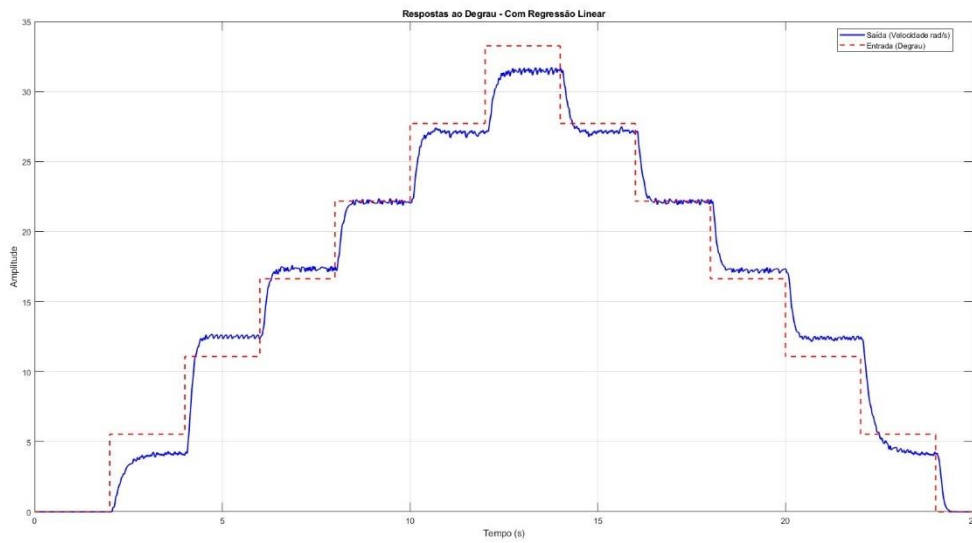
uma melhora significativa na estabilidade relativa nessas condições, reduzindo os impactos adversos do atrito e garantindo um desempenho mais robusto em situações críticas.

Figura 32 – Resposta em Malha Aberta sem Regressão



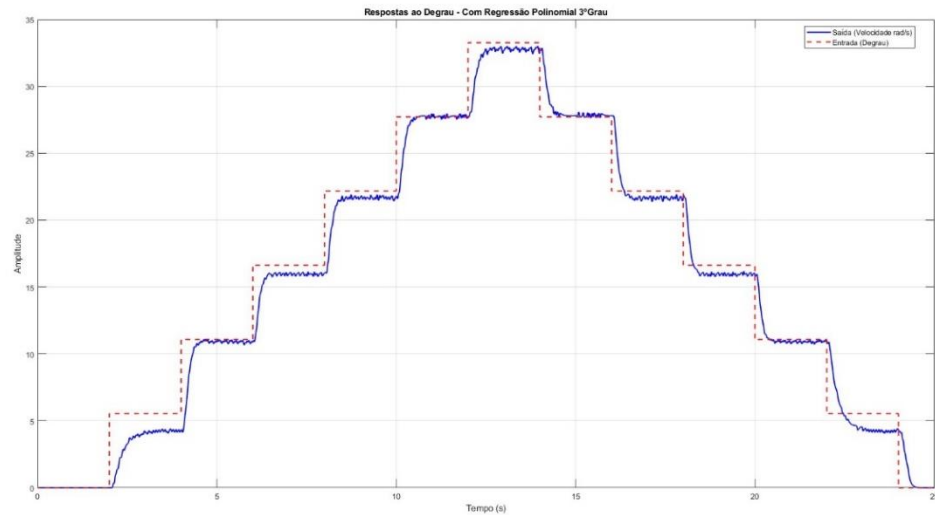
Fonte: do Autor

Figura 33 – Resposta em Malha Aberta com Regressão Linear



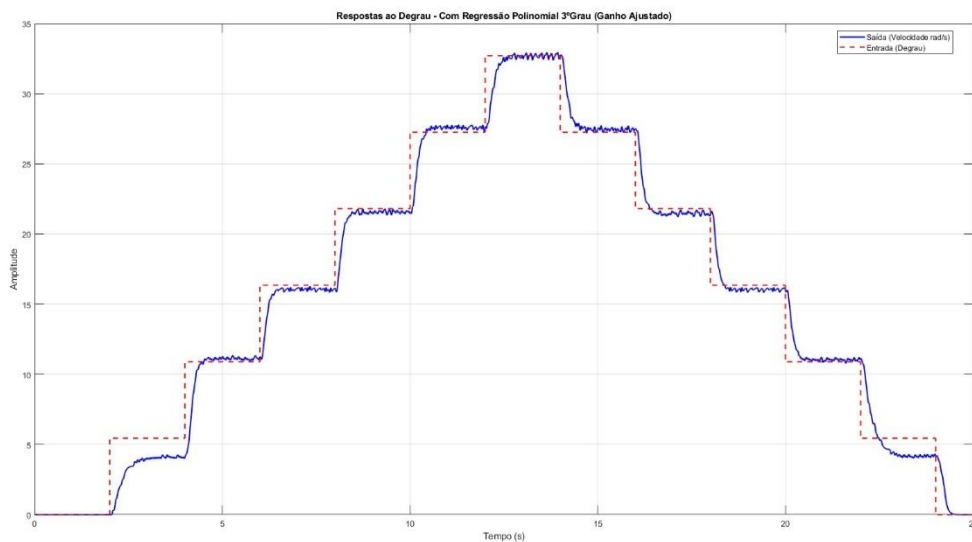
Fonte: do Autor

Figura 34 – Resposta em Malha Aberta com Regressão Polinomial de 3ª Ordem



Fonte: do Autor

Figura 35 – Resposta em Malha Aberta com Regressão Polinomial de 3ª Ordem com Ganho Ajustado.



Fonte: do Autor

Percebe-se uma melhora significativa na representação do sistema ao aplicar a regressão, o que impacta diretamente na qualidade do controle em malha fechada. Isso ocorre porque as respostas passam a apresentar características transitórias mais consistentes. Considerando que o **controle clássico** se baseia em **funções de transferência lineares**, esse comportamento é essencial para uma aplicação eficaz.

## **4 Resultados**

Neste capítulo, são apresentados os resultados das análises realizadas. Serão implementadas as respostas em malha fechada utilizando os controladores projetados, analisando seu desempenho e verificando se as características dinâmicas obtidas estão próximas da realidade do sistema.

Além disso, ressalta-se a melhoria significativa obtida após os ajustes nos modelos, destacando o impacto positivo no desempenho do controle e comprovando a viabilidade de aplicar o controle clássico de forma eficaz, o que constitui o principal diferencial deste trabalho. Por fim, aborda-se o impacto da metodologia no aprendizado dos alunos, enfatizando como ela contribuiu para um entendimento mais aprofundado do funcionamento dos sistemas de controle.

### **4.1 Análise da Resposta Dinâmica do Sistema Comparando os Modelos**

Nesta seção, são apresentados e analisados os resultados referentes à resposta dinâmica do sistema, comparando os modelos matemático e experimental desenvolvidos. O objetivo é avaliar a precisão com que cada modelo representa o comportamento real do sistema sob diferentes condições de entrada. Para isso, são utilizadas as respostas em malha fechada, com análise detalhada das características transitórias e de regime permanente.

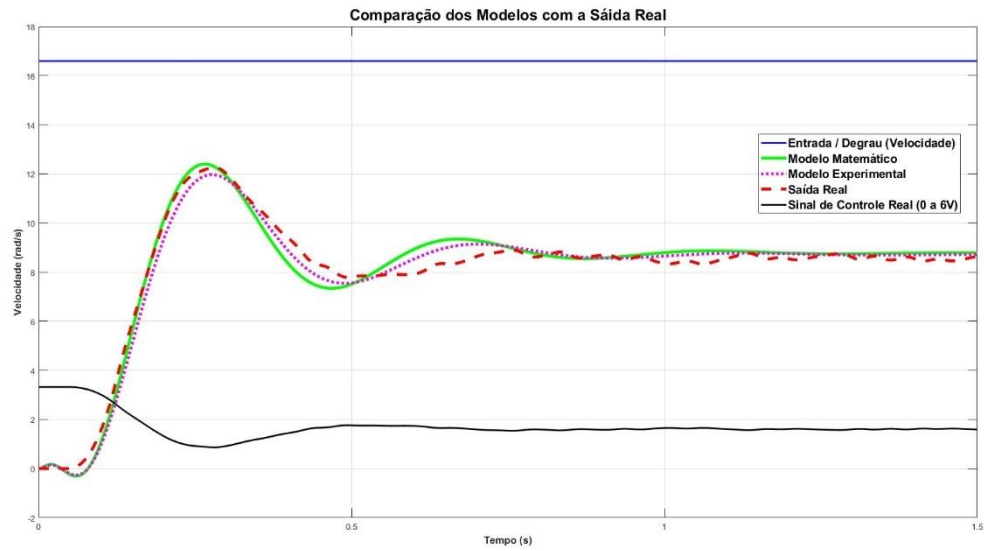
#### **4.1.1 Entrada Degrau**

A análise detalhada da resposta do sistema à entrada degrau, com foco na avaliação das características da resposta transitória, permite identificar parâmetros fundamentais, como tempo de subida, tempo de acomodação, overshoot e estabilidade, que são cruciais para a avaliação do desempenho do sistema. Os resultados obtidos possibilitam a comparação entre o comportamento previsto pelo modelo matemático e o desempenho observado experimentalmente, verificando a aderência entre teoria e prática.

Essa abordagem também serve para validar a eficiência dos controladores projetados e a representatividade dos modelos matemáticos, garantindo que eles capturam com precisão as dinâmicas do sistema real. Ao fim, a análise busca assegurar que os controladores e modelos empregados atendam aos requisitos de desempenho e estabilidade desejados para a aplicação.

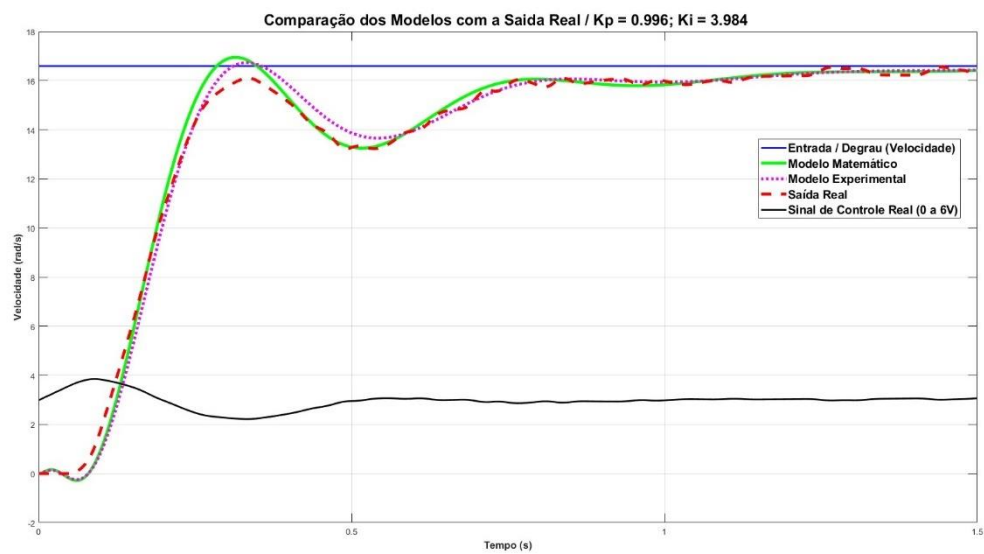
- Saída(rad/s)
- Entrada(V)

Figura 36 – Resposta em Malha Fechada - **P** - Degrau (16.59 rad/s);  $K_p = 1.107$



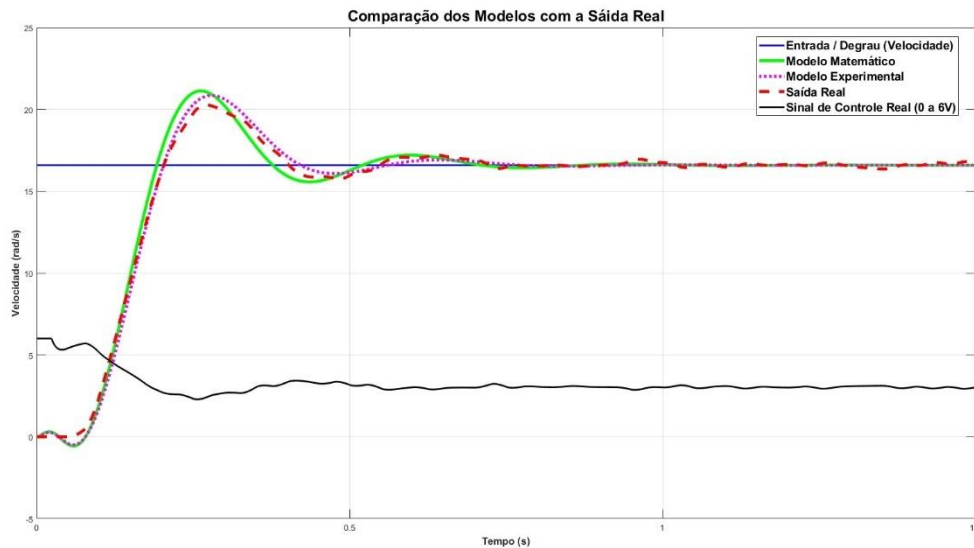
Fonte: do Autor

Figura 37 – Resposta em Malha Fechada - **PI** - Degrau (16.59 rad/s)



Fonte: do Autor

Figura 38 – Resposta em Malha Fechada - PID - Degrau (16.59 rad/s)



Fonte: do Autor

Ao analisar os resultados obtidos com o controle em malha fechada, observa-se que a resposta que mais se aproximou do comportamento real do sistema foi a representada pela curva de cor rosa, derivada do modelo experimental. Essa curva, associada à função de transferência  $G_2(s)$  do modelo experimental, demonstrou ser a que melhor caracteriza o sistema em estudo para as condições de entrada analisadas.

Na Figura 36, é possível identificar a presença de um erro em estado estacionário, que pode ser calculado pela fórmula:

$$e_{ss} = \frac{1}{1 + \lim_{s \rightarrow 0} (G_1(s) \cdot H(s))} \quad (4.1)$$

$$e_{ss} = \frac{1}{1 + \lim_{s \rightarrow 0} \frac{5.45}{0.083s + 1} \cdot 0.1835} \quad (4.2)$$

$$e_{ss} = 50\% \quad (4.3)$$

Essa análise é confirmada graficamente, já que o erro em estado estacionário mostrado na Figura 36 aproxima-se de fato de 50%. Além disso, outros aspectos dinâmicos do sistema também se mostram consistentes. O tempo de acomodação e o overshoot, que representam características transitórias fundamentais do sistema, estão muito próximos dos valores reais observados. Essa correspondência reforça a capacidade do modelo em reproduzir com precisão a dinâmica dominante do sistema.

Nas Figuras 37 e 38, os resultados das outras respostas seguem o mesmo padrão: as especificações de desempenho apresentaram uma semelhança impressionante com os dados reais. Esse fato é notável, considerando os desafios envolvidos no projeto de sistemas de controle. Não apenas foi possível alcançar a estabilidade, o que por si só já é um objetivo desafiador, mas também atender simultaneamente às especificações de desempenho dinâmico e estacionário. Essa conquista é ainda mais significativa ao se considerar que foi obtida utilizando apenas técnicas de controle clássico.

Esse resultado destaca a eficácia e a robustez do modelo experimental na representação do sistema, demonstrando que, mesmo com ferramentas convencionais, é possível alcançar um controle preciso e estável, atendendo aos requisitos de projeto com grande eficiência. Essa combinação de precisão e estabilidade é essencial em aplicações práticas, onde a tolerância a erros e a resposta a perturbações são fatores críticos para o sucesso do projeto.

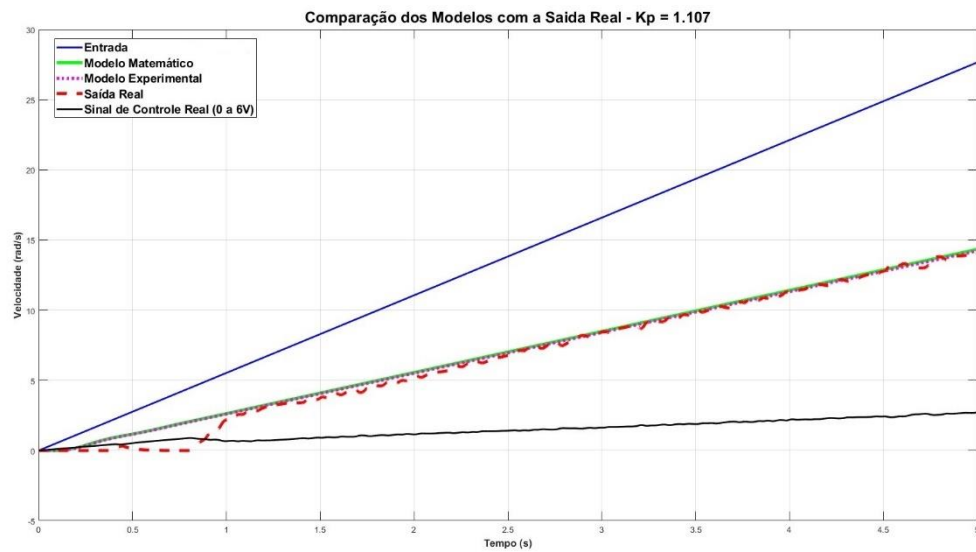
Quando os professores ensinarem controle aos alunos e eles tiverem a chance de projetar um controlador que funcione, isso aumentará o interesse pela disciplina. Ver na prática a aplicação dos conceitos teóricos torna o aprendizado mais significativo e envolvente, mostrando que o controle vai além de equações abstratas e pode resolver problemas reais, motivando os estudantes a se aprofundarem na área.

#### **4.1.2 Entrada Rampa**

A análise da resposta do sistema à entrada rampa visa avaliar o comportamento dinâmico do sistema frente a uma variação linear da entrada com o tempo. Esse tipo de entrada é particularmente útil para estudar a resposta do sistema a mudanças contínuas e graduais, simulando condições mais realistas de operação, como ajustes progressivos em um processo ou controle de velocidade. A resposta transitória é observada para verificar o tempo de estabilização, a precisão do seguimento da rampa e o erro de regime permanente.

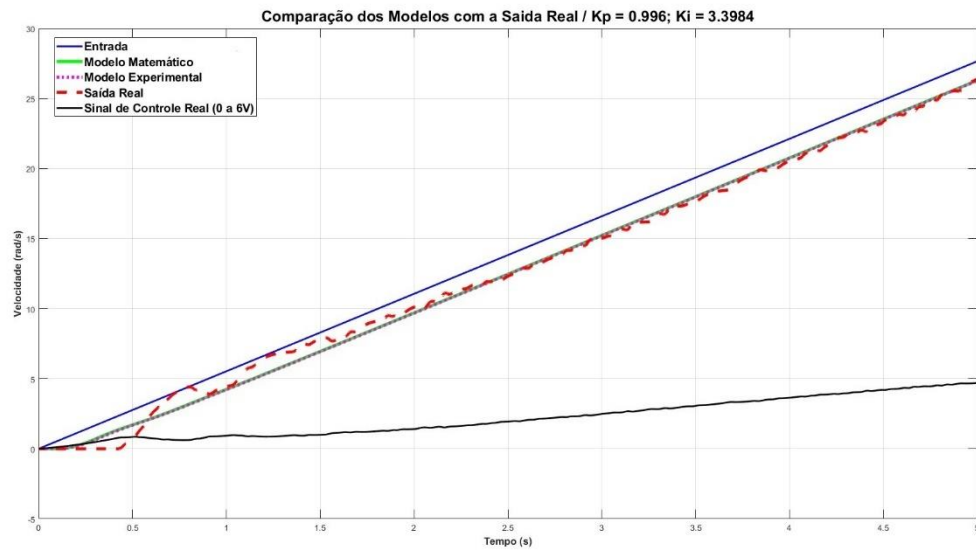
A comparação entre os resultados obtidos pelos modelos matemáticos e os dados experimentais permite avaliar a eficácia dos controladores e a representatividade dos modelos no contexto da entrada rampa. Essa análise é essencial para validar se o sistema pode responder de forma eficiente a variações constantes da entrada, mantendo a precisão e o desempenho desejados.

Figura 39 – Resposta em Malha Fechada - **P** - Rampa (0 a 27,25 rad/s)



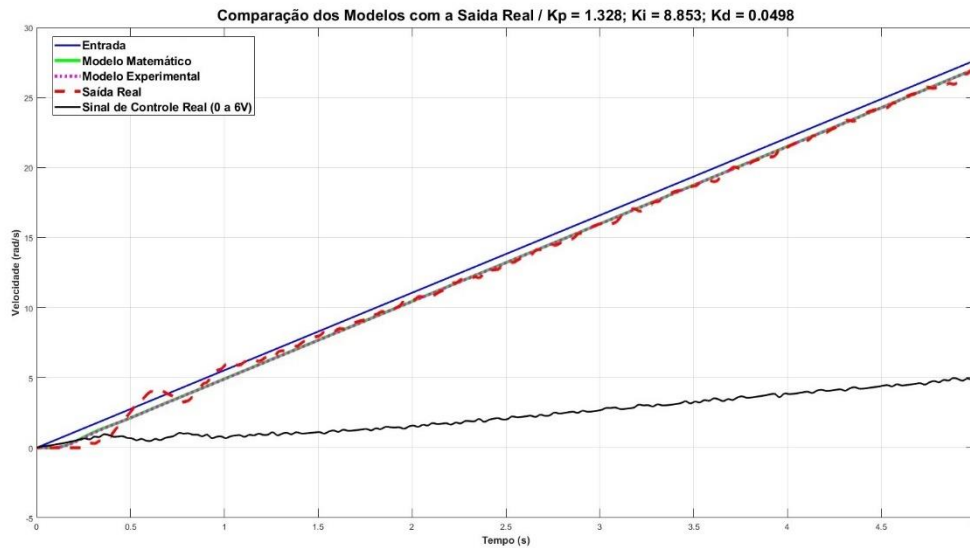
Fonte: do Autor

Figura 40 – Resposta em Malha Fechada - **PI** - Rampa (0 a 27,25 rad/s)



Fonte: do Autor

Figura 41 – Resposta em Malha Fechada - PID - Rampa (0 a 27,25 rad/s)



Fonte: do Autor

Ao analisar a resposta do sistema à entrada em rampa sob controle em malha fechada, observa-se que o modelo experimental, representado pela curva rosa associada à função de transferência  $G_2(s)$ , demonstrou um excelente desempenho ao aproximar-se do comportamento real do sistema. Esse resultado é particularmente significativo, pois a entrada em rampa explora uma **faixa mais ampla** de operação do sistema, desafiando a linearidade assumida nos modelos. Geralmente, sistemas reais apresentam comportamentos não lineares que dificultam o acompanhamento fiel de respostas desse tipo, tornando ainda mais impressionante a precisão alcançada pelo modelo.

Uma observação importante é que, quando o motor precisa superar a inércia inicial, o modelo não acompanha perfeitamente o comportamento real do sistema. No entanto, uma vez que o motor sai dessa condição inicial e entra em operação contínua, o modelo se ajusta e consegue seguir a resposta do sistema de maneira notavelmente fiel. Isso demonstra a robustez do modelo em capturar a dinâmica do sistema mesmo em cenários desafiadores, adaptando-se às condições transitórias para alcançar uma boa correspondência com o comportamento real.

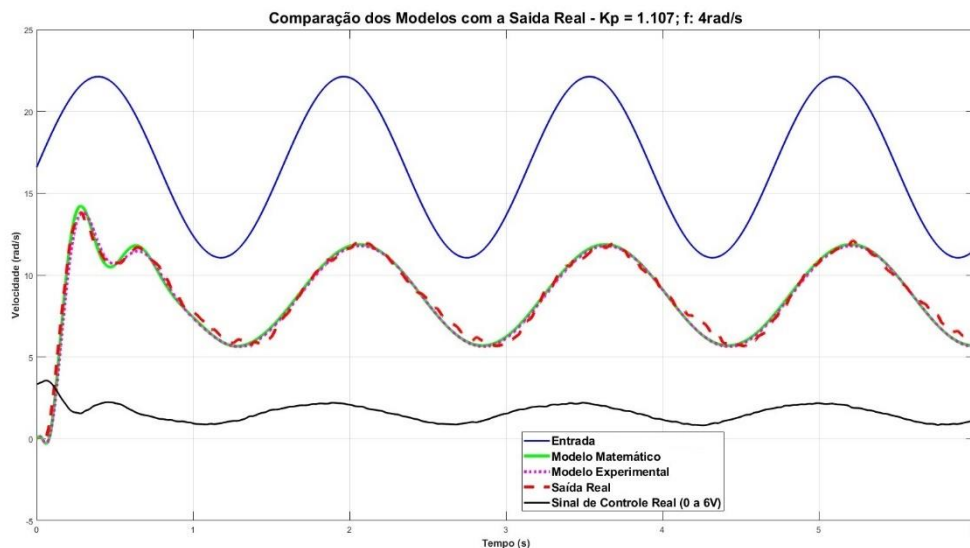
As Figuras 39, 40 e 41 reforçam a qualidade dessa aproximação, evidenciando especificações de desempenho dinâmico e estacionário que refletem com precisão a realidade. Esse nível de aderência é notável, especialmente considerando as dificuldades inerentes ao controle de sistemas não lineares. A capacidade de um modelo linearizado de representar adequadamente uma resposta em rampa indica uma sólida identificação dos parâmetros e uma modelagem cuidadosa.

### 4.1.3 Entrada Senoidal

A análise da resposta do sistema à entrada senoidal tem como objetivo investigar o comportamento dinâmico do sistema quando submetido a uma variação periódica e oscilatória da entrada. A entrada senoidal é uma excelente representação de sinais alternados ou oscilantes, sendo amplamente utilizada para avaliar a resposta em regimes de frequência e testar a linearidade e estabilidade do sistema. Ao aplicar essa entrada, observam-se características como a amplitude da resposta, o desfasamento entre a entrada e a saída, e a capacidade do sistema de seguir corretamente a oscilação, identificando possíveis ressonâncias ou instabilidades.

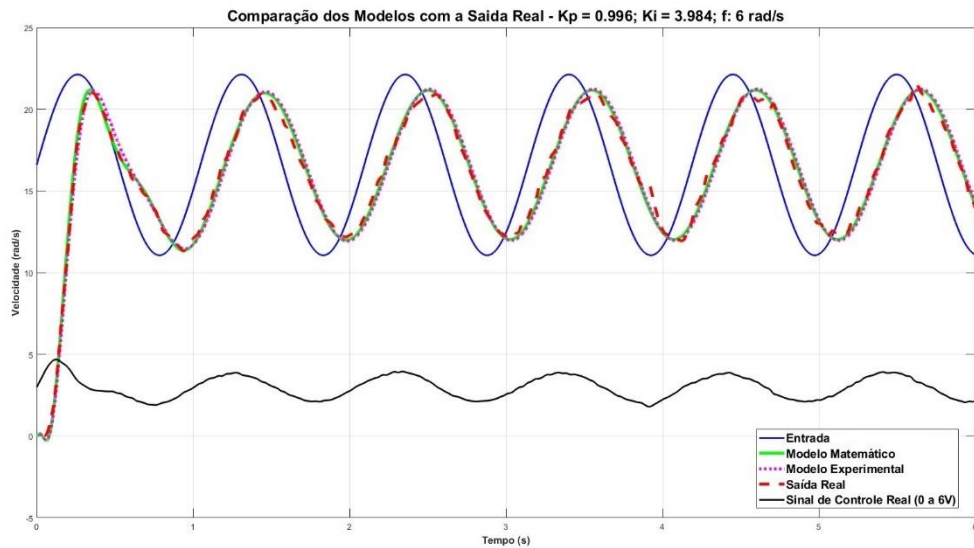
A comparação entre a resposta prevista pelo modelo matemático e os dados experimentais permite verificar a acuracidade dos modelos e a eficiência dos controladores na mitigação de distúrbios e na manutenção da amplitude e fase desejadas. Além disso, a análise da entrada senoidal é crucial para entender como o sistema se comporta em frequências específicas, sendo fundamental na validação do desempenho em sistemas de controle que operam com sinais periódicos. Essa avaliação também permite identificar possíveis limitações do sistema, como a resposta de frequência, e ajustar os parâmetros dos controladores para otimizar o seguimento do sinal senoidal e garantir uma operação estável e precisa.

Figura 42 – Resposta em Malha Fechada - P - Entrada Senoidal (Velocidade)



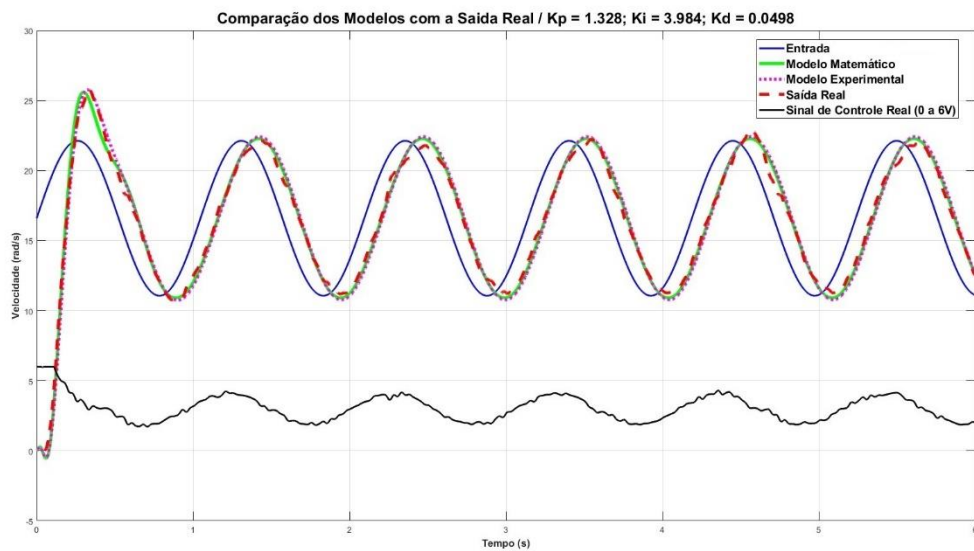
Fonte: do Autor

Figura 43 – Resposta em Malha Fechada - **PI** - Entrada Senoidal (Velocidade)



Fonte: do Autor

Figura 44 – Resposta em Malha Fechada - **PID** - Entrada Senoidal (Velocidade)



Fonte: do Autor

A análise das Figuras 42, 43 e 44 apresenta o desempenho do sistema em malha fechada sob uma entrada senoidal, considerando a aplicação de três controladores diferentes: proporcional (P), proporcional-integral (PI) e proporcional-integral-derivativo (PID). Além disso, observa-se que tanto o sistema real quanto os dois modelos desenvolvidos acompanham

a saída de maneira muito precisa, reforçando a qualidade dos modelos e dos controladores aplicados.

Na Figura 42, é analisado o comportamento do sistema utilizando o controlador proporcional. Esse tipo de controlador ajusta a saída do sistema com base na magnitude do erro, resultando em uma resposta que acompanha a entrada senoidal, mas com limitações. É comum que haja pequenas diferenças de fase e amplitude, especialmente em frequências mais altas, já que o controlador proporcional não compensa integralmente os atrasos inerentes à dinâmica do sistema. Apesar disso, o acompanhamento geral da entrada senoidal é satisfatório, demonstrando um bom controle básico do sistema.

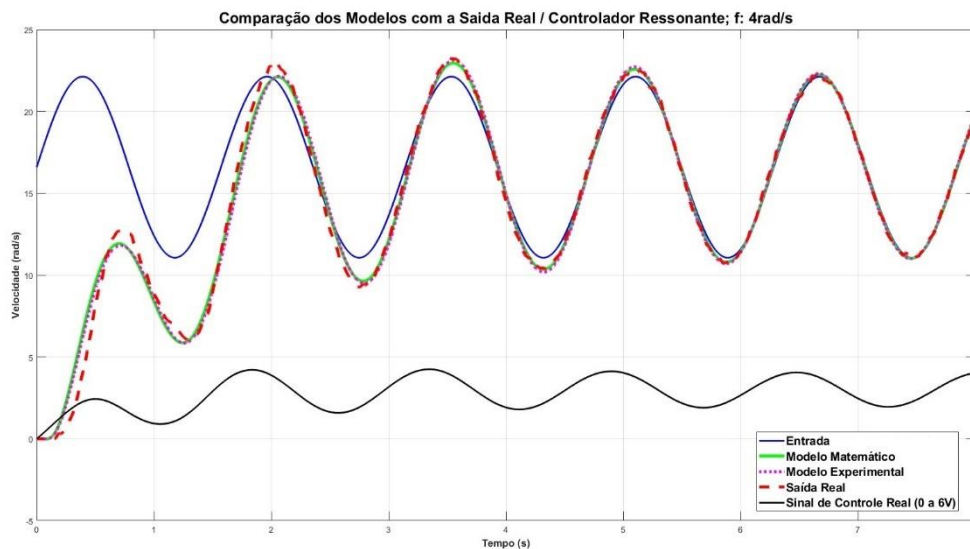
A Figura 43 mostra o impacto do controlador proporcional-integral (PI). Com a inclusão da ação integral, o sistema ganha a capacidade de eliminar erros em regime permanente, ajustando-se melhor à amplitude da entrada senoidal. No entanto, a ação integral, por sua natureza, pode introduzir um atraso adicional na resposta, aumentando a diferença de fase em relação ao sinal de entrada. Mesmo com essa característica, a saída do sistema mantém-se bem próxima da entrada senoidal, refletindo a eficiência do controlador PI em corrigir desvios acumulados ao longo do tempo.

Na Figura 44, observa-se o efeito do controlador proporcional-integral-derivativo (PID). A adição da ação derivativa ao controle melhora significativamente o desempenho do sistema, reduzindo tanto as oscilações quanto as diferenças de fase e amplitude. A ação derivativa antecipa mudanças no erro, contribuindo para uma resposta mais rápida e precisa às variações da entrada senoidal. Como resultado, a saída do sistema com o controlador PID apresenta o melhor acompanhamento da entrada, com uma correspondência mais fiel em termos de amplitude e menor atraso de fase.

O desempenho do sistema em malha fechada foi analisado com três tipos de controladores: proporcional (P), proporcional-integral (PI) e proporcional-integral-derivativo (PID), sob uma entrada senoidal. Tanto o sistema real quanto os modelos propostos apresentaram um bom acompanhamento da saída, destacando a eficácia dos controladores. A análise das Figuras 42, 43 e 44 mostrou as características de cada controlador. O controlador proporcional da Figura 42 resultou em um bom controle básico, embora com limitações em altas frequências. O controlador PI da Figura 43 melhorou a precisão na amplitude da entrada, mas introduziu um pequeno atraso de fase devido à ação integral. Já o controlador PID da Figura 44 proporcionou o melhor desempenho, com menor diferença de fase e amplitude, devido à ação derivativa, que antecipou as variações no erro.

Em resumo, o controlador PID se destacou pela capacidade de melhorar a resposta do sistema, oferecendo maior precisão no acompanhamento da entrada senoidal, superando as limitações dos controladores proporcional e proporcional-integral.

Figura 45 – Resposta em Malha Fechada - Controlador Ressonante



Fonte: do Autor

A Figura 45 apresenta o sistema com um controlador ressonante somado a um integrador, resultando em rastreamento perfeito da entrada senoidal, com diferença nula entre entrada e saída em regime permanente. O controlador ressonante que pode ser observado na Equação 4.4, atua amplificando a resposta em uma frequência específica, definida no denominador do controlador, que deve coincidir exatamente com a frequência do sinal de entrada para garantir o rastreamento.

$$C(s) = \frac{s}{s^2 + \omega^2} + \frac{1}{s} \quad (4.4)$$

A inclusão do integrador reforça essa capacidade, corrigindo desvios acumulados ao longo do tempo e eliminando erros. Observa-se que tanto o sistema real quanto os modelos seguem a entrada senoidal com precisão, sem diferenças de amplitude ou fase, demonstrando a eficácia dessa configuração em sistemas com entradas periódicas.

A entrada senoidal utilizada nas figuras é uma ferramenta valiosa para avaliar as características dinâmicas do sistema, pois explora uma ampla faixa de operação, destacando comportamentos que podem ser desafiadores para modelos e controladores. Os resultados demonstram uma clara evolução do desempenho à medida que o controle avança do

proporcional para o proporcional-integral e, finalmente, para o proporcional-integral-derivativo. Essa progressão ressalta como os diferentes controladores afetam a precisão e a capacidade de acompanhar a entrada senoidal, evidenciando a eficácia dos ajustes implementados.

Embora seja comum esperar desvios significativos entre o modelo e o comportamento real para entradas senoidais, devido à complexidade das dinâmicas envolvidas, os ajustes realizados neste trabalho garantiram que os controladores clássicos apresentassem um desempenho muito satisfatório. As respostas acompanharam de forma notavelmente próxima o sistema real, demonstrando que, mesmo com limitações inerentes ao controle clássico, foi possível alcançar resultados robustos e representativos do comportamento do sistema em condições dinâmicas variadas.

## 4.2 Impacto dos Ajustes no Desempenho do Sistema

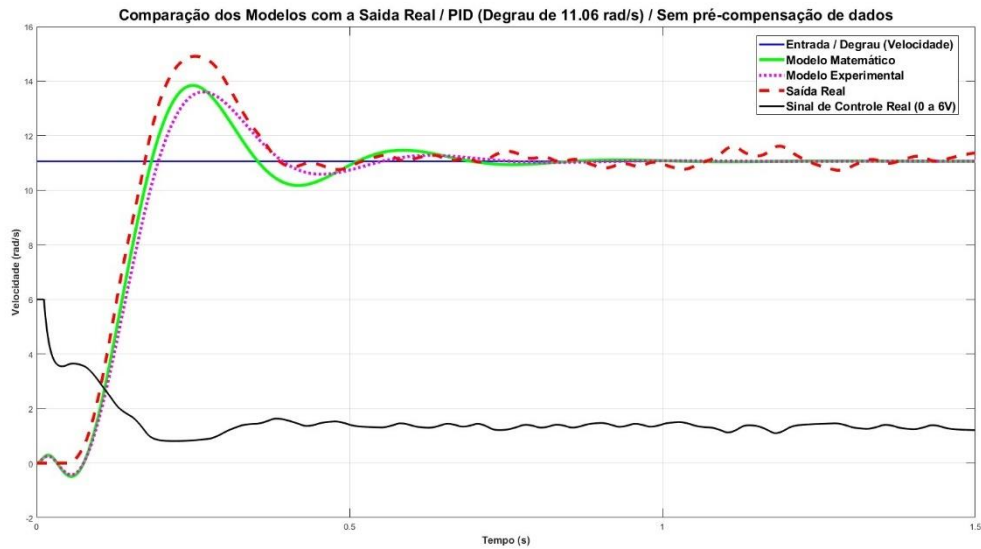
A análise dos impactos gerados pelos **ajustes** no sistema, resultantes da **pré-compensação de dados**, visa identificar as melhorias proporcionadas por essa técnica. A pré-compensação de dados, ao representar o sistema por meio de polinômios de ordem superior, permite um maior controle sobre as características da resposta transitória, o que resulta em uma maior precisão na previsão e no gerenciamento do comportamento do sistema.

A implementação do ajuste por meio da regressão polinomial de terceira ordem trouxe melhorias significativas no desempenho do sistema em malha fechada. Na Figura 46, que ilustra a resposta do sistema sem o ajuste, foi observado um comportamento não desejado, incluindo um overshoot elevado que não estava previsto no projeto. Esse overshoot foi consequência direta das não linearidades presentes na ponte H e da relação entre o PWM gerado pelo Arduino e a tensão efetivamente aplicada ao motor. Essas não linearidades introduziram variações nos ganhos do sistema, resultando em uma resposta imprevisível e divergente do modelo teórico. Em ambos os gráficos foram aplicados **os mesmos controladores PID**.

Após o ajuste realizado pela pré-compensação dos dados, conforme apresentado na Figura 47, essas discrepâncias foram minimizadas. A regressão polinomial permitiu modelar a relação não linear entre o PWM e a saída da ponte H, possibilitando que o Arduino realizasse uma **pré-compensação** antes de enviar o sinal ao motor. Essa abordagem ajustou o comportamento da planta, como podemos observar na Figura 36. Com isso, foi possível obter uma resposta com overshoot reduzido, tempos de subida e acomodação mais estáveis e um **comportamento transitório mais previsível**. Essas melhorias foram essenciais para alinhar o

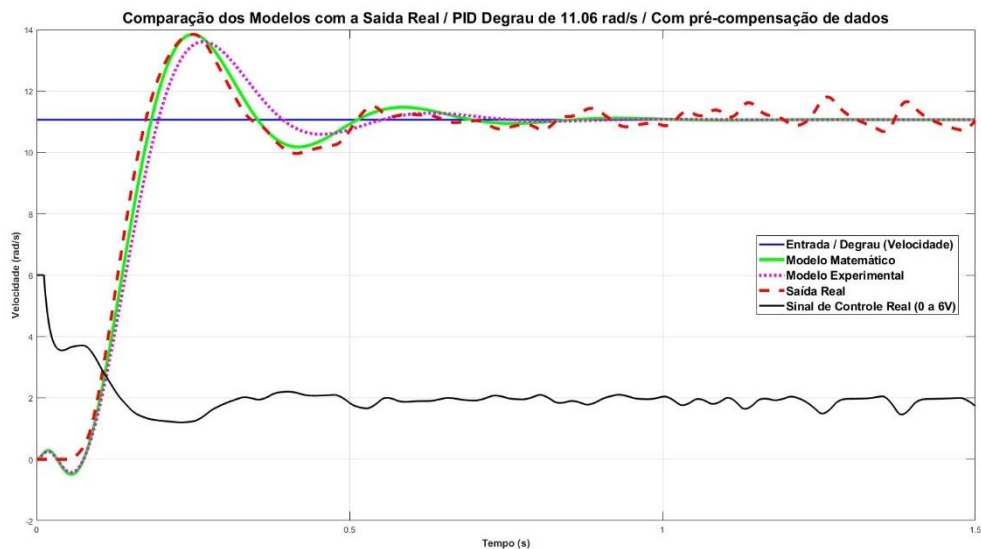
comportamento do sistema real ao modelo projetado, garantindo maior precisão e eficiência no controle.

Figura 46 – Resposta em Malha Fechada PID - Sem Ajustes



Fonte: do Autor

Figura 47 – Resposta em Malha Fechada PID - Com Ajustes



Fonte: do Autor

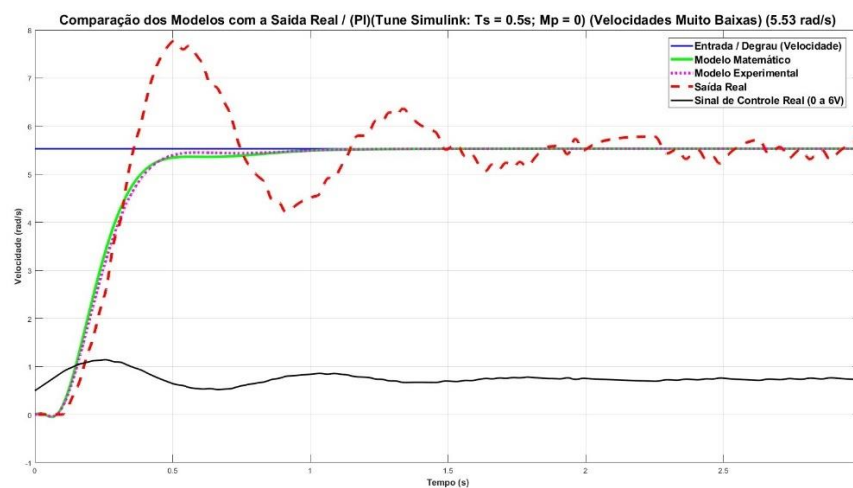
A pré-compensação de dados foi especialmente importante para a **aplicação do controle clássico**, que é o tema central deste trabalho. O controle clássico, como o PID utilizado, assume que a planta do sistema é linear. Sem o ajuste realizado, as **variações de ganho** introduzidas

pela **não linearidade** da ponte H comprometeriam o desempenho do controlador, levando a respostas inconsistentes e divergentes em diferentes condições de operação. Com o ajuste, foi possível garantir que a planta **atendesse aos requisitos do modelo teórico**, permitindo que o controlador clássico trabalhasse de forma confiável e previsível.

Além disso, o sucesso dessa abordagem dependeu de alguns cuidados fundamentais. A coleta de dados precisos foi essencial para criar um modelo de regressão confiável, representando adequadamente a relação entre o PWM e a tensão de saída. A escolha da regressão polinomial de terceira ordem mostrou-se apropriada para capturar as não linearidades sem introduzir complexidade excessiva, enquanto a validação do modelo assegurou sua precisão em toda a faixa de operação. Esses cuidados garantiram que o ajuste fosse eficaz e que o controle clássico pudesse ser aplicado com confiança.

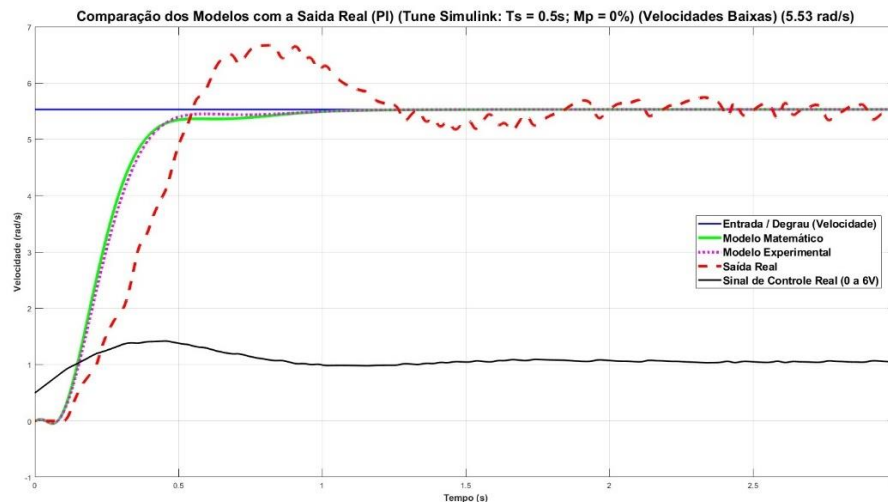
Em conclusão, a regressão polinomial desempenhou um papel crítico na melhoria do desempenho do sistema, alinhando o comportamento da planta com os pressupostos do controle clássico. A Figura 47 demonstra que, com o ajuste, o sistema respondeu conforme o esperado, enquanto a Figura 46 destaca os problemas que ocorreriam sem esse tratamento das não linearidades. Sem esse ajuste, o controle clássico enfrentaria variações significativas, comprometendo a estabilidade e a previsibilidade do sistema. Isso ressalta a importância de tratar as não linearidades para garantir o sucesso de abordagens clássicas no controle de sistemas reais.

Figura 48 – Resposta em Malha Fechada PI - Sem Ajustes



Fonte: do Autor

Figura 49 – Resposta em Malha Fechada PI - COM Ajustes



Fonte: do Autor

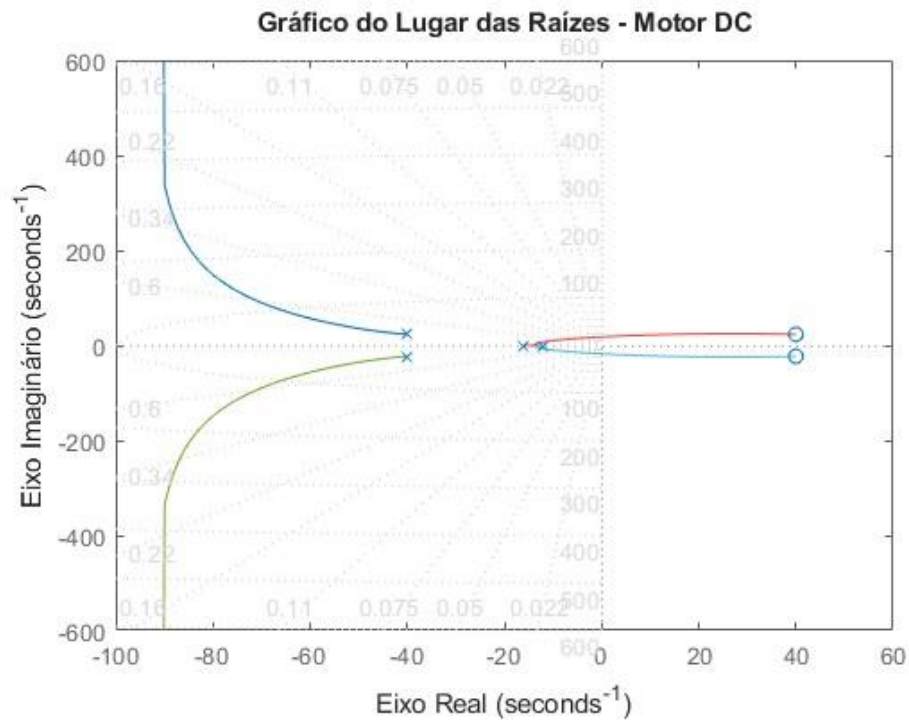
Nas Figuras 48 e 49, é possível observar o comportamento do sistema em velocidades baixas. Nessa faixa, o atrito aumenta consideravelmente, o que faz com que o sistema apresente um comportamento significativamente diferente. No entanto, percebe-se que, com a aplicação da pré-compensação de dados, embora não tenha sido possível atender rigorosamente às características da resposta transitória, houve uma melhoria significativa na estabilidade relativa do sistema. Isso resultou em uma redução das oscilações, tanto em número quanto em amplitude, proporcionando um desempenho mais estável e controlado.

### 4.3 Lugar das Raízes e Diagrama de Bode

O Gráfico do Lugar das Raízes que pode ser observado na Figura 50 é uma ferramenta essencial no ensino de controle clássico, pois oferece uma visualização intuitiva do comportamento dos polos do sistema conforme o ganho do controlador é ajustado. Na aplicação prática com o motor DC, utilizando Arduino e Simulink, o LGR permite demonstrar conceitos importantes como a estabilidade do sistema.

A posição dos polos no plano complexo indica a estabilidade: polos no semiplano esquerdo garantem estabilidade, enquanto polos no semiplano direito indicam instabilidade. A partir desse gráfico, os alunos podem acompanhar a trajetória dos polos à medida que o ganho varia no sistema real observando o gráfico, analisando como o deslocamento desses polos influencia diretamente a estabilidade do sistema.

Figura 50 – Lugar das Raízes



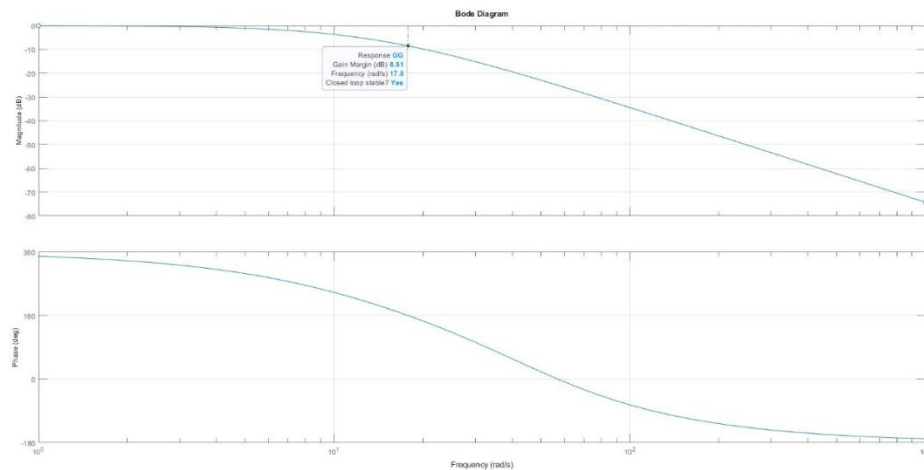
Fonte: do Autor

Além disso, o LGR permite explicar a relação entre a parte real e a parte imaginária dos polos, mostrando como esses valores influenciam no tempo de resposta do sistema. O deslocamento dos polos em direção ao eixo real, por exemplo, indica uma resposta mais rápida, porém com menor oscilação. Isso facilita o entendimento prático dos conceitos de tempo de acomodação e frequência de oscilação.

Outra aplicação relevante do LGR está no projeto de controladores, um tópico amplamente explorado na disciplina de Controle II. A partir da função de transferência do sistema, que pode apresentar certa complexidade, os alunos têm a oportunidade de projetar diferentes tipos de controladores. Essa abordagem torna o aprendizado mais dinâmico e envolvente, permitindo que os estudantes validem, na prática, se as especificações de desempenho foram atendidas, transformando o processo em uma experiência educativa e motivadora.

Por meio dessa abordagem, os estudantes conseguem compreender de forma mais concreta as equações teóricas de controle e a dinâmica de sistemas físicos. A visualização direta torna o aprendizado mais claro e prático, integrando teoria e experimentação no estudo de controle clássico.

Figura 51 – Diagrama de Bode



Fonte: do Autor

O Diagrama de Bode que pode ser observado na Figura 51 é outra ferramenta fundamental no ensino de controle clássico, pois possibilita a análise da resposta em frequência do sistema. Na prática com o motor DC controlado pelo Arduino e Simulink, o Bode é essencial para ensinar conceitos como margem de ganho e margem de fase. O diagrama superior, que mostra a magnitude do sistema em função da frequência, permite identificar a frequência de ganho unitário e analisar a margem de ganho, que está diretamente relacionada à estabilidade do sistema. Já o diagrama inferior, que apresenta a variação da fase, possibilita a determinação da margem de fase, fornecendo informações sobre a robustez do sistema frente a variações de ganho.

Além da análise de estabilidade, o Diagrama de Bode é amplamente utilizado no projeto de **compensadores** de avanço e atraso de fase. Os alunos podem visualizar como a adição de um compensador afeta a resposta em frequência e, conseqüentemente, melhora a estabilidade e a performance do sistema. O gráfico permite também a associação entre a resposta em frequência e o comportamento transitório do sistema no domínio do tempo, ensinando os alunos a correlacionar conceitos como faixa passante, frequência de corte e tempo de resposta.

A combinação dessas ferramentas teóricas com a implementação prática usando Simulink e Arduino cria um ambiente de aprendizado dinâmico e motivador. Os alunos têm a oportunidade de observar, na prática, como os modelos matemáticos representam sistemas reais e como as estratégias de controle influenciam o comportamento do sistema. Esse tipo de abordagem promove uma aprendizagem ativa, permitindo que os conceitos abstratos de

controle clássico, muitas vezes desafiadores, sejam compreendidos de forma concreta e aplicada.

Para realizar todas as análises de controle clássico e observar os conceitos de forma prática em um sistema real, os alunos podem utilizar a função de transferência completa do sistema que pode ser obtida na Figura 30. Essa função inclui a parte de primeira ordem, o ganho, o atraso e a dinâmica do filtro. Dessa forma, a função de transferência do sistema é representada da seguinte forma:

$$G(s) = \frac{88.48s^2 - 7078s + 1.887 \cdot 10^5}{0.083s^4 + 8.968s^3 + 379.3s^2 + 6246s + 34128} \cdot 0.1808 \quad (4.5)$$

#### 4.4 Impacto Educacional e Aprendizagem dos Alunos

A proposta apresentada neste trabalho teve como principal objetivo oferecer aos alunos uma experiência prática e interativa no estudo de controle clássico, algo que raramente é demonstrado de forma funcional no ambiente acadêmico. Na realidade da faculdade em que o estudo foi desenvolvido, os conceitos de controle são frequentemente abordados apenas de maneira teórica, sem que os alunos tenham a oportunidade de visualizar o funcionamento prático dos controladores aplicados em sistemas reais. Essa lacuna muitas vezes gera frustração, especialmente porque, na prática, os sistemas reais não apresentam um comportamento completamente linear, o que pode resultar em respostas que não condizem com as especificações teóricas projetadas.

A ausência de linearidade nos sistemas testados poderia levar os alunos a concluir que as equações e os métodos clássicos aprendidos na disciplina de controle não são aplicáveis na prática, o que impactaria negativamente sua motivação e interesse pela área. No entanto, o ajuste realizado neste trabalho, por meio da regressão polinomial de terceira ordem, permitiu representar melhor o sistema com polinômios de ordem superior, conseqüentemente, alinhar a resposta prática do sistema ao modelo teórico. Dessa forma, os alunos podem projetar controladores clássicos, como proporcional (P), proporcional-integral (PI) e proporcional-integral-derivativo (PID), utilizando métodos como a curva de reação ou qualquer outro método clássico, e obter resultados práticos consistentes com suas previsões teóricas.

Esse alinhamento entre teoria e prática é um fator essencial para aumentar o interesse dos alunos pela disciplina. Ao constatarem que os cálculos realizados levam a resultados práticos previsíveis, os alunos percebem a aplicabilidade dos conceitos aprendidos e são incentivados a explorar mais profundamente a área de controle. Além disso, com a função de

transferência do sistema encontrada, é possível que eles projetem e implementem outros controladores, como compensadores de avanço e atraso de fase, ou utilizem métodos baseados em diagramas de Bode, lugar das raízes ou análise na frequência.

Outro aspecto relevante é a simplicidade da implementação no Simulink, que facilita a interação dos alunos com os controladores projetados. Ao encontrarem os parâmetros de controle e simplesmente adicionarem os blocos correspondentes no Simulink, eles podem observar a resposta do sistema de forma intuitiva e direta. Isso não apenas reforça o aprendizado dos conceitos teóricos, mas também promove a interação com os professores e aumenta a eficácia das metodologias ativas aplicadas na disciplina.

O impacto educacional também se estende à versatilidade do sistema proposto. O uso de uma protoboard, em vez de uma placa fixa, foi pensado para oferecer aos alunos uma plataforma adaptável. Isso permite que o sistema seja facilmente modificado para diferentes aplicações, como o controle de sistemas de equilíbrio, por exemplo, um sistema ball-and-beam. Os mesmos princípios apresentados neste trabalho podem ser utilizados para modelar outros sistemas, projetar controladores e visualizar as respostas dinâmicas no Simulink.

Figura 52 – Conceitos de Controle na Prática



Fonte: do Autor

Na Figura 52, foi realizada uma explicação detalhada para os alunos sobre como interpretar o diagrama de Bode. Para isso, foram aplicadas diversas frequências no sistema em malha aberta, utilizando o motor, variando desde as frequências mais baixas até as mais altas. Durante o experimento, os alunos observaram o gráfico de saída da velocidade, notando que, à

medida que as frequências aumentavam, a amplitude das oscilações diminuía, e o gráfico apresentava um deslocamento.

Em outras palavras, eles estavam percebendo que o ganho em dB do sistema estava diminuindo, ao mesmo tempo em que a fase do gráfico estava sendo deslocada. Esse comportamento observado pelos alunos é exatamente o que o diagrama de Bode da Figura 51 ilustra: a relação entre a frequência de operação e as variações de ganho e fase de um sistema dinâmico. Esse exercício prático proporcionou uma compreensão mais clara de como o sistema responde a diferentes frequências e reforçou os conceitos teóricos relacionados ao diagrama de Bode, tornando algo abstrato e difícil em uma experiência iterativa e concreta.

## 5 Conclusões

O trabalho desenvolvido demonstrou a viabilidade e a importância de integrar ferramentas práticas ao ensino de controle clássico, promovendo uma experiência que une teoria e prática de forma consistente. A abordagem proposta não apenas permitiu a validação dos conceitos de controle clássico em um sistema físico real, mas também solucionou as discrepâncias comuns entre o comportamento teórico e o comportamento prático dos sistemas, especialmente em decorrência das não linearidades.

A aplicação de uma regressão polinomial para ajustar a planta foi um elemento crucial para alinhar o sistema à modelagem teórica, permitindo respostas previsíveis e consistentes com os parâmetros projetados. Dessa forma, os alunos puderam vivenciar de maneira concreta a aplicabilidade dos métodos clássicos de controle, utilizando ferramentas como o Simulink para validar suas projeções e observar o funcionamento dos controladores implementados.

Essa metodologia provou ser eficaz em aumentar o interesse dos alunos pela área de controle, oferecendo uma experiência mais interativa e significativa. Além disso, o trabalho reforçou a importância de ferramentas acessíveis, como o Arduino, para tornar o aprendizado de conceitos complexos mais intuitivo e prático. A combinação de modelagem teórica, ajustes para lidar com a não linearidade e implementação prática proporcionou uma abordagem integrada e pedagógica para o ensino de controle clássico.

### 5.1 Trabalhos Futuros

Este trabalho abre diversas possibilidades para estudos e desenvolvimentos futuros, tanto na área de controle quanto na aplicação educacional. Uma das principais sugestões é a adaptação do sistema para diferentes plantas físicas, como o controle de sistemas ball-and-beam, de temperatura, ou mesmo sistemas robóticos simples. A flexibilidade oferecida pelo uso da protoboard permite que os alunos explorem novas aplicações, ampliando o alcance do conhecimento adquirido.

Outra possibilidade é a incorporação de técnicas mais avançadas de controle, como controle robusto, adaptativo ou mesmo o uso de controladores baseados em aprendizado de máquina, para complementar os métodos clássicos. Essa expansão proporcionaria aos alunos uma visão mais abrangente da área de controle, conectando fundamentos teóricos a abordagens contemporâneas

Por fim, recomenda-se a criação de materiais didáticos baseados neste trabalho, incluindo tutoriais e guias para o uso do Simulink e do Arduino no contexto do controle clássico. Esses materiais poderiam ser integrados às disciplinas de controle, ampliando ainda mais o impacto educacional do projeto e contribuindo para uma formação prática mais sólida na engenharia.

## Referências

RECK, Rebecca M.; SREENIVAS, Ramavarapu S. Developing a new affordable DC motor laboratory kit for an existing undergraduate controls course. In: 2015 American Control Conference (ACC). IEEE, 2015. p. 2801-2806. Citado na página **1**.

FERREIRA, Ana LS et al. O Problema da Defasagem entre a Teoria e a Prática: Proposta de uma Solução de Compromisso para um Problema Clássico de Controle. In: Anais: XXXIV–Congresso Brasileiro de Ensino de Engenharia. Passo Fundo: Universidade de Passo Fundo. 2006. Citado na página **2**.

BARBOSA, Gabriel Nogueira; CARDOSO, Lucas Pimenta. Desenvolvimento de uma bancada didática para as disciplinas de sistemas de controle com identificação de planta e projeto de controlador PID via simulação computacional. 2020. Citado nas páginas **2, 3 e 4**.

CAMPOS, Andrei Henrique Patriota et al. Estratégias de controle aplicadas a um atuador de liga de memória de forma para o controle de deformação de uma viga engastada. Dissertação de Mestrado. Curso de Engenharia Elétrica, Universidade Federal de Campina Grande, Campina Grande, 2018. Citado nas páginas **31, 32 e 33**.

BARROSO, Antônio Júlio Santana. Caracterização, modelagem e controle de um atuador de liga de memória de forma. 2019. Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal de Campina Grande, Campina Grande, 2019. Citado nas páginas **44 e 45**.

PINTO, Jan Erik Mont Gomery. Aplicação prática do método de sintonia de controladores PID utilizando o método do relé com histerese. 2014. Dissertação de Mestrado. Universidade Federal do Rio Grande do Norte. Citado nas páginas **20, 21, 23 e 24**

ŠKRABA, Andrej; STANOVVOV, Vladimir; SEMENKIN, Eugene. Development of control systems kit for study of PID controller in the framework of cyber-physical systems. In: IOP Conference Series: Materials Science and Engineering. IOP Publishing, 2020. p. 012105. Citado na página **4**.

DE LIMA, Alequine Batista; BARROS, Péricles Rezende; JÚNIOR, George Acioli. Módulo didático para ensino de teoria de controle. In: Congresso Brasileiro de Automática-CBA. 2019. Citado na página **3**.

FRANKLIN, Gene F.; POWELL, J. David; EMAMI-NAEINI, Abbas. Sistemas de controle para engenharia. Bookman Editora, 2013. Citado na página **77**.

COOK, Jeffrey A.; SAMAD, Tariq. Controls curriculum survey. IEEE Control Systems Society Outreach Task Force Report, 2009. Citado na página **12**.

HERADIO, Ruben et al. Virtual and remote labs in education: A bibliometric analysis. Computers & Education, v. 98, p. 14-38, 2016. Citado na página **15**.

LUCENA, Simone. Culturas digitais e tecnologias móveis na educação<sup>1</sup>. Educar em Revista, p. 277-290, 2016. Citado na página **275**.

MURRAY, M. R. Future directions on control, dynamics and systems report. Control in an Information Rich World, California Institute of Technology, 2002. Citado na página **8**.

AGUIRRE, HENRIQUE DE ABREU RANGEL. Implementação de controle digital de posição do motor CC em tempo real utilizando MATLAB e Simulink. Citado nas páginas **26, 27 e 28**.

NERI, Hermes Gustavo Fernandes. Utilização da plataforma Arduino para controle de experimentos remotos de física. 2014. Citado na página **23**.

OGATA, Katsuhiko. *Engenharia de controle moderno*. 5. ed. São Paulo: Pearson Prentice Hall, 2010. Citado nas páginas **11, 15, 16, 17 e 62**.

MAYA, P. A.; LEONARDI, F. *Controle essencial*. 2. ed. São Paulo: Pearson Education do Brasil, 2014. Citado na página **22**.

NISE, N. S. *Engenharia de sistemas de controle*. 7. ed. Rio de Janeiro: LTC, 2017. Citado na página **44**.

DORF, R.C; BISHOP, R.H. Sistemas de controle modernos. 13. ed. Rio de Janeiro: LTC, 2018. Citado na página **35**.

KLUEVER, Craig A. Sistemas dinâmicos: modelagem, simulação e controle. Rio de Janeiro: LTC, 2017. Citado na página **137**.

ABRANTES, Camila de Oliveira et al. Experimentos didáticos para identificação e controle PID utilizando ferramenta de geração automática de código. 2019. Citado nas páginas **25, 26 e 27**.

ROBOCORE. Módulo Driver Ponte H - L298N. 2024. Disponível em: <[https:// www.robocore.net/driver-motor/driver-motor-ponte-h-l298n](https://www.robocore.net/driver-motor/driver-motor-ponte-h-l298n)>. Citado na página **47**.

AUTOCOREROBOTICA. Motor DC 6V 280 RPM com Roda e Encoder. 2024. Disponível em: <<https://www.autocorerobotica.com.br/motor-dc-6v-280-rpm-com-roda-e-encoder>> Acesso em: 02.12.2024. Citado na página **1**.

SEED. JGA25-370 Geared Motor. 2024. Disponível em: <[https://media.digikey.com/pdf/Data%20Sheets/Seed%20Technology/114090046\\_Web.pdf](https://media.digikey.com/pdf/Data%20Sheets/Seed%20Technology/114090046_Web.pdf)>. Acesso: 02.12.2024 Citado nas páginas **1 e 2**.

RIOLINK. Fonte Variável Yaxun 1502dd+ 15v 2a alimentação110v. 2024. <<https://www.riolink.com.br/instrumentos-de-teste-e-medicao/fonte-de-alimentacao/fonte-variavel-com-rf-paracelular-yaxun-1502dd-15v-2a-110v>>. Acesso: 02.12.2024 Citado na página **1**.

CULTURA, B. Arduino, o documentário. 2010. Via upload no vimeo.com. Disponível em: <<https://vimeo.com/31389230>>. Acesso em: 15.06.2023. Citado na página **21**.