

**UFMS**

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL

FACULDADE DE ENGENHARIAS, ARQUITETURA E URBANISMO E GEOGRAFIA

CURSO DE ENGENHARIA ELÉTRICA

**Internet das Coisas (IoT) aplicado no controle online de  
variação de velocidade de um motor de indução**

**Gabriel Santos Rodrigues**

Campo Grande - MS

9 de dezembro de 2024

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL  
FACULDADE DE ENGENHARIAS, ARQUITETURA E URBANISMO E GEOGRAFIA  
CURSO DE ENGENHARIA ELÉTRICA

## **Internet das Coisas (IoT) aplicado no controle online de variação de velocidade de um motor de indução**

**Gabriel Santos Rodrigues**

Trabalho de Conclusão de Curso apresentado  
como exigência para obtenção do grau de  
Bacharelado em Engenharia Elétrica da  
Universidade Federal de Mato Grosso do Sul –  
UFMS.

Orientador: Raymundo Cordero Garcia

Campo Grande - MS  
9 de dezembro de 2024

# **Internet das Coisas (IoT) aplicado no controle online de variação de velocidade de um motor de indução**

Monografia apresentada à Universidade Federal de Mato Grosso do Sul na Faculdade de  
Engenharias, Arquitetura e Urbanismo e Geografia, para obtenção da Graduação em  
Engenharia Elétrica.

Banca Examinadora:

---

Prof. Dr. Raymundo Cordero Garcia  
Orientador

---

Prof. Dr. Luigi Galotto Junior

---

MSc. Wisam Ali Mohamad Issa

Campo Grande - MS  
9 de dezembro de 2024

## DECLARAÇÃO DE AUTORIA E RESPONSABILIDADE

Eu, Gabriel Santos Rodrigues, residente e domiciliado na cidade de Campo Grande, Estado do Mato Grosso do Sul, declaro que o Trabalho de Conclusão de Curso (TCC) apresentado, com o título “**Internet das Coisas (IoT) aplicado no controle online de variação de velocidade de um motor de indução**” é de minha autoria e assumo a total responsabilidade pelo seu conteúdo e pela originalidade do texto. Declaro que identifiquei e referenciei todas as fontes e informações gerais que foram utilizadas para construção do presente texto. Declaro também que este artigo não foi publicado, em parte, na íntegra ou conteúdo similar em outros meios de comunicação, tendo sido enviado com exclusividade para a Universidade Federal de Mato Grosso do Sul (UFMS).

-----

**Gabriel Santos Rodrigues**

*Dedico este trabalho a Deus, que me concedeu  
sabedoria e discernimento para concluí-lo.  
Senhor, que este seja apenas o início de muitas  
conquistas sob suas bênçãos.*

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, por me conceder saúde para concluir este trabalho.

Agradeço a minha família, que muito me apoiou, não somente no desenvolvimento desse trabalho, mas durante todo o desenrolar do curso de engenharia. À minha mãe, Iara, por sempre acreditar no meu potencial. Ao meu pai, Dario, por me apoiar financeiramente e emocionalmente em todas as ocasiões. A minha irmã, Wis, pelo apoio constante. Sem vocês, nada disso seria possível.

À minha namorada, Rayane, que esteve ao meu lado, encorajando-me positivamente na conclusão deste trabalho.

Agradeço aos meus amigos, Hyan e Matheus, por me acompanharem em cada etapa deste percurso, e por sempre demonstrarem interesse pela minha pesquisa. Graças a vocês, o desenvolvimento desse curso foi mais leve.

Por fim, registro meu agradecimento especial ao meu orientador, Dr. Raymundo Cordero Garcia, que desde a Iniciação Científica me apoia incansavelmente. Sua paciência e dedicação pelo ensino são admiráveis, e serviram como fonte de inspiração para este projeto.

## RESUMO

A Internet das Coisas (IoT) pode ser definida como a conexão entre o mundo físico e o virtual, envolvendo equipamentos inteligentes com capacidade de sensoriamento, processamento e comunicação digital.

Projetos na área de IoT cresceram de forma significativa na última década, uma vez que oferecem inúmeras possibilidades, tanto no ambiente residencial, como controle remoto de luminosidade, comandos por voz e inteligências artificiais, quanto no setor industrial, com controle de velocidade de motores e monitoramento de equipamentos.

O presente trabalho tem como objetivo elaborar um sistema de controle de operação de um variador de velocidade para motor de indução trifásico usando Internet das Coisas e um microcontrolador ESP32. Foi desenvolvido um servidor WEB, no ESP32, que pode ser acessado através de uma conexão Wi-Fi com um smartphone ou computador. Este servidor WEB oferece ao usuário uma interface que permite selecionar a frequência elétrica de operação do variador de velocidade. O ESP recebe a referência e produz um sinal PWM com um ciclo de trabalho proporcional à frequência elétrica desejada. Um filtro passa-baixas converte o sinal PWM em um sinal analógico que define a frequência de operação do variador.

O microcontrolador ESP32 foi desenvolvido como uma solução eficiente para projetos na área de IoT, possibilitando diversas aplicações por meio de conexões sem fio, como Wi-Fi e Bluetooth. Além disso, sua programação pode ser realizada na plataforma Arduino IDE, o que se mostra uma solução prática e acessível. Adicionalmente, este trabalho constitui uma aplicação prática de diversos conteúdos aprendidos ao longo do curso de Engenharia Elétrica.

Os resultados experimentais indicam que o sistema de controle baseado permite um controle adequado da velocidade do motor, com as vantagens oferecidas pela IoT.

**Palavras-Chave:** Internet das Coisas, ESP32, Controle de Velocidade, Servidor Web.

## ABSTRACT

The Internet of Things (IoT) can be defined as the connection between the physical and virtual worlds, involving intelligent devices with sensing, processing, and digital communication capabilities.

Projects in the IoT field have grown significantly over the last decade, as they offer numerous possibilities in both residential environments, such as remote light control, voice commands, and artificial intelligence, and industrial applications, such as motor speed control and equipment monitoring.

The objective of this work is to develop a control system for the operation of a variable speed drive for a three-phase induction motor using the Internet of Things and an ESP32 microcontroller. It was developed a WEB server, not ESP32, which can be accessed through a Wi-Fi connection with a smartphone or computer. This WEB server offers the user an interface that allows selecting the electrical operating frequency of the variable speed drive. The ESP receives a reference and produces a PWM signal with a duty cycle proportional to the desired electrical frequency. A low-pass filter converts the PWM signal into an analog signal that defines the operating frequency of the drive.

The ESP32 microcontroller was developed as an efficient solution for projects in the IoT area, enabling various applications through wireless connections, such as Wi-Fi and Bluetooth. Also, your programming can be carried out on the Arduino IDE platform, or a practical and accessible solution is shown. Additionally, this work constitutes a practical application of various contents learned throughout the Electrical Engineering course.

The experimental results indicate that the based control system allows adequate control of the motor speed, as well as the advantages offered by the IoT.

**Keywords:** Internet of Things, ESP32, Controlling the speed, Web server.

---

## LISTA DE FIGURAS

|                                                                                                                                                                       |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1 - Gráfico de interesse ao longo do tempo na pesquisa do termo "IoT" no mecanismo de pesquisa Google, no Mundo. (período de 01.01.2004 a 24.10.2024).....     | 14 |
| Figura 2 - AmpOp elementar.....                                                                                                                                       | 20 |
| Figura 3 - Diagrama de bloco elementar de um LIT .....                                                                                                                | 21 |
| Figura 4 - Diagrama de Bode de um polo simples. ....                                                                                                                  | 22 |
| Figura 5 - Diagrama de Bode de um zero simples.....                                                                                                                   | 23 |
| Figura 6 - Gráfico com a resposta em frequência do ganho de um filtro PB ideal. ....                                                                                  | 25 |
| Figura 7 - Circuito RC de um filtro passa baixas passivo de 1ª ordem.....                                                                                             | 25 |
| Figura 8 - Circuito de um filtro passa baixas ativo de 2ª ordem com ganho unitário, de Sallen-Key.....                                                                | 26 |
| Figura 9 - Respostas de filtros de 2ª ordem com diferentes valores de Q.....                                                                                          | 28 |
| Figura 10 - Resposta em frequência da magnitude de filtros Butterworth, de 1ª, 2ª, 4ª e 10ª ordem. ....                                                               | 29 |
| Figura 11 - Pinagem do microcontrolador ESP32.....                                                                                                                    | 30 |
| Figura 12 - Microcontrolador ESP32.....                                                                                                                               | 31 |
| Figura 13 - Gráfico de interesse ao longo do tempo na pesquisa do termo "ESP32" no mecanismo de pesquisa Google, no mundo. (período de 06.09.2016 a 15.10.2024). .... | 31 |
| Figura 14 - BNDES Pilotos IoT – 3 áreas priorizadas. ....                                                                                                             | 33 |
| Figura 15 - Blocos básicos de IoT.....                                                                                                                                | 34 |
| Figura 16 - Exemplo básico da estrutura de um arquivo HTML5. ....                                                                                                     | 36 |
| Figura 17 - Exemplo básico de um arquivo CSS3.....                                                                                                                    | 37 |
| Figura 18 - Estrutura básica de um arquivo Javascript.....                                                                                                            | 38 |
| Figura 19 - Página web simples criada. ....                                                                                                                           | 38 |
| Figura 20 - Diagrama de motores elétricos.....                                                                                                                        | 39 |
| Figura 21 - Visão em corte de um motor elétrico de indução. ....                                                                                                      | 40 |
| Figura 22 - Rotor gaiola de esquilo.....                                                                                                                              | 41 |
| Figura 23 - Rotor bobinado. ....                                                                                                                                      | 41 |
| Figura 24 - Enrolamentos trifásicos do estator. ....                                                                                                                  | 42 |
| Figura 25 - Circuito equivalente por fase de um motor de indução.....                                                                                                 | 44 |

|                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------|----|
| Figura 26 - Diagrama do fluxo de potência de um motor de indução. ....                                          | 44 |
| Figura 27 - Circuito equivalente do motor de indução - Teorema de Thevenin.....                                 | 45 |
| Figura 28 - Curva característica de conjugado x velocidade.....                                                 | 46 |
| Figura 29 - Controle de velocidade pela tensão de linha.....                                                    | 47 |
| Figura 30 - Diagrama básico de um inversor de frequência.....                                                   | 49 |
| Figura 31 - Curva característica da tensão em função da frequência de alimentação – Padrão de<br>uso geral..... | 50 |
| Figura 32 – Diagrama geral do sistema proposto. ....                                                            | 51 |
| Figura 33 - ESP32 do BATLAB.....                                                                                | 52 |
| Figura 34 - Circuito interno.....                                                                               | 53 |
| Figura 35 - AmpOp TL082 do BATLAB.....                                                                          | 53 |
| Figura 36 - Inversor de frequência do BATLAB.....                                                               | 54 |
| Figura 37 - Bornes de potência do inversor do BATLAB. ....                                                      | 55 |
| Figura 38 - Bornes de controle do inversor do BATLAB.....                                                       | 56 |
| Figura 39 - Diagrama do inversor. ....                                                                          | 57 |
| Figura 40 - Motor de indução do BATLAB. ....                                                                    | 57 |
| Figura 41 - Especificações técnicas do motor de indução.....                                                    | 58 |
| Figura 42 - Interface do servidor web. ....                                                                     | 60 |
| Figura 43 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 0Hz.....                                         | 60 |
| Figura 44 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 30Hz.....                                        | 61 |
| Figura 45 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 60Hz.....                                        | 61 |
| Figura 46 - Diagrama elétrico do filtro PB de 2ª ordem de Butterworth com topologia Sallen-Key.<br>.....        | 63 |
| Figura 47 - Diagrama de Bode da resposta em frequência do filtro. ....                                          | 63 |
| Figura 48 - Diagrama elétrico do circuito amplificador não inversor.....                                        | 64 |
| Figura 49 - Sinal de tensão de saída filtrado e amplificado – 0Hz.....                                          | 65 |
| Figura 50 - Sinal de tensão de saída filtrado e amplificado – 30Hz.....                                         | 65 |
| Figura 51 - Sinal de tensão de saída filtrado e amplificado – 60Hz.....                                         | 66 |
| Figura 52 - Bornes do controle de tensão analógica do inversor. ....                                            | 66 |
| Figura 53 - Diagrama completo do sistema proposto.....                                                          | 67 |
| Figura 54 - Sistema completo montado no laboratório BATLAB.....                                                 | 67 |

---

## LISTA DE TABELAS

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| Tabela 1 - Ganho de potência linear, em Bel e em decibel. ....           | 18 |
| Tabela 2 - Ganho de tensão linear, em Bel e em decibel.....              | 19 |
| Tabela 3 - Coeficientes de filtros de 2ª ordem. ....                     | 27 |
| Tabela 4 – Especificações do ESP8266 e ESP32 .....                       | 30 |
| Tabela 5 - Características principais do AmpOp tl082.....                | 53 |
| Tabela 6 - Descrições dos bornes de potência do inversor do BATLAB. .... | 55 |
| Tabela 7 - Descrições dos bornes de potência do inversor do BATLAB. .... | 56 |

---

## SUMÁRIO

|          |                                            |           |
|----------|--------------------------------------------|-----------|
| <b>1</b> | <b>INTRODUÇÃO.....</b>                     | <b>14</b> |
| 1.1      | Contextualização.....                      | 14        |
| 1.2      | Objetivos .....                            | 15        |
| 1.3      | Organização do Trabalho.....               | 16        |
| <b>2</b> | <b>FUNDAMENTAÇÃO TEÓRICA.....</b>          | <b>17</b> |
| 2.1      | Teoria dos filtros analógicos .....        | 17        |
| 2.1.1    | Decibel (dB).....                          | 17        |
| 2.1.2    | Amplificadores operacionais.....           | 19        |
| 2.1.3    | Resposta Em Frequência .....               | 21        |
| 2.1.4    | Diagrama de Bode .....                     | 22        |
| 2.1.5    | Filtros Analógicos.....                    | 23        |
| 2.2      | Esp-32 .....                               | 29        |
| 2.2.1    | Rede Wi-fi .....                           | 32        |
| 2.3      | Internet das coisas (IoT) .....            | 32        |
| 2.4      | HTML .....                                 | 35        |
| 2.4.1    | CSS E JAVASCRIPT .....                     | 36        |
| 2.5      | Motor de indução .....                     | 38        |
| 2.5.1    | Princípio de funcionamento.....            | 42        |
| 2.5.2    | Característica torque-velocidade.....      | 44        |
| 2.5.3    | Controle de velocidade .....               | 46        |
| 2.5.4    | Inversor de Frequência.....                | 48        |
| <b>3</b> | <b>SISTEMA PROPOSTO .....</b>              | <b>51</b> |
| 3.1      | Arduino IDE e Visual Studio Code .....     | 51        |
| 3.2      | Microcontrolador ESP32 do BATLAB.....      | 52        |
| 3.3      | Circuito de filtragem e amplificação ..... | 53        |
| 3.4      | Inversor de frequência do BATLAB .....     | 54        |
| 3.5      | Motor de indução do BATLAB.....            | 57        |

---

|          |                                  |           |
|----------|----------------------------------|-----------|
| <b>4</b> | <b>RESULTADOS.....</b>           | <b>59</b> |
| 4.1      | Primeira etapa .....             | 59        |
| 4.2      | Segunda etapa .....              | 62        |
| 4.3      | Terceira etapa .....             | 64        |
| 4.4      | Quarta etapa.....                | 66        |
| <b>5</b> | <b>conclusões .....</b>          | <b>68</b> |
| <b>6</b> | <b>REFERÊNCIAS .....</b>         | <b>69</b> |
| <b>7</b> | <b>APÊNDICES.....</b>            | <b>74</b> |
| 7.1      | Programação no Arduino IDE ..... | 74        |
| 7.2      | Programação no Matlab.....       | 77        |

# 1 INTRODUÇÃO

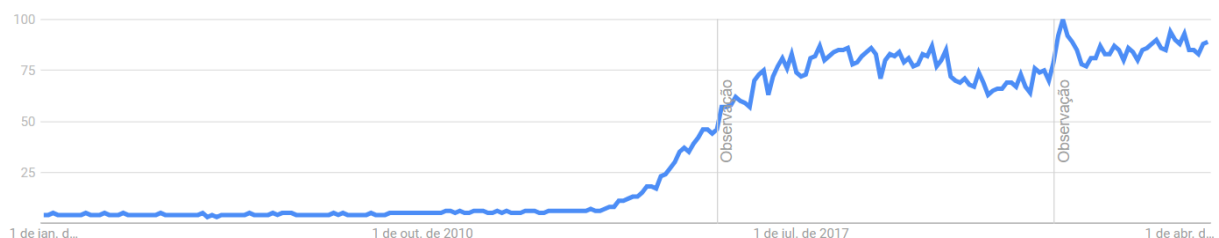
## 1.1 CONTEXTUALIZAÇÃO

A inovação tecnológica proporcionada pelo advento da internet contribuiu diretamente para o desenvolvimento de tecnologias presentes em nosso cotidiano. Computadores e celulares “smartphones”, assistentes de voz inteligente como Amazon Alexa e Google Assistente, além do surgimento de tecnologias de compartilhamento e armazenamento de dados, como a computação em nuvem a adoção dos protocolos de internet. Diante disso, um termo emergiu: IoT (“*Internet of Things*” – Internet das Coisas) (DR.P.B.PATHAK, 2016).

IoT foi um termo inicialmente proposto pelo britânico Kevin Ashton, professor do MIT (“*Massachusetts Institute of Technology*” – Instituto de Tecnologia de Massachusetts), presente no título de sua apresentação sobre RFID (“*Radio Frequency Identification*” - “Identificadores de Rádio Frequência”), realizada na empresa P&G (Procter & Gamble), em 1999 (RFID JOURNAL, 2009). Trata-se de uma tecnologia emergente, pois contribuiu para um avanço importante na área da tecnologia, assim, permite que diversos equipamentos ao nosso redor possam se comunicar entre si, por meio de uma rede Wi-fi (SHULONG WANG et al., 2016).

A justificativa para elaboração de um projeto na área de IoT, tem como motivação a popularidade e atualidade desse tema, tendo em vista a grande quantidade de projetos e soluções desenvolvidas com essa abordagem nos dias de hoje. Pela figura 1, pode-se perceber o interesse crescente no conceito de IoT.

Figura 1 - Gráfico de interesse ao longo do tempo na pesquisa do termo "IoT" no mecanismo de pesquisa Google, no Mundo. (período de 01.01.2004 a 24.10.2024).



Fonte: (GOOGLE TRENDS, [s.d.])

---

A proposta desse trabalho, vem como uma alternativa para o acionamento de um motor de indução trifásico, onde o seu controle de velocidade ocorre remotamente, por intermédio de uma página web no formato HTML, com a possibilidade de um controle online variável, com o microcontrolador ESP32 como servidor do sistema. O desenvolvimento desse projeto foi motivado pelo seu baixo custo e complexidade, podendo ser aprimorado em futuras aplicações.

Projetos de acionamento e monitoramento de motores, através de um módulo Wi-fi, com possibilidade de controle remoto, são temas atuais e populares entre os projetos em IoT. É o caso do projeto de pesquisa em (ELOI; FRANÇA; CAMPOS, 2023).

Assim como esse trabalho, alguns projetos implementam soluções de baixo custo baseados no microcontrolador ESP32, através do monitoramento de motores de indução trifásicos, com o acréscimo da aquisição de dados de medição de corrente, tensão e temperatura, com a possibilidade de envio dos dados para a nuvem (RIOS, 2019).

Alguns outros trabalhos exploram o uso da tecnologia dos Gêmeos Digitais, como forma de monitorar motores elétricos industriais, com o foco na manutenção preditiva, evitando danos maiores ao sistema, e na otimização do desempenho, gerando eficiência (ALEXANDRE ABRAHAMI PINTO DA CUNHA, 2023).

Em suma, estudos e projetos desenvolvidos na área de IoT, mostram-se inovadores e práticos, além da sua viabilidade de implantação aliada ao seu baixo custo, com a possibilidade de expansão do sistema para aquisição de dados mais complexos.

## **1.2 OBJETIVOS**

O objetivo geral deste trabalho é implementar um sistema de controle de velocidade de um motor de indução trifásico através de uma interface web interativa com controle variável de frequência, utilizando o microcontrolador ESP32.

Os objetivos específicos são:

- Criar um servidor web com HTML5.
- Instalar o microcontrolador ESP32 no Arduino IDE.
- Implementar o controle de frequência proporcional a modulação PWM (Pulse Width Modulation – Modulação por Largura de Pulso) do ESP32.
- Implementar filtro passa-baixa no sinal de saída do ESP32.

- 
- Integrar o sinal de saída do ESP32 ao inversor de frequência.
  - Teste prático com o motor de indução.

### **1.3 ORGANIZAÇÃO DO TRABALHO**

O capítulo 1 indica o contexto do trabalho, o problema a ser tratado e os objetivos do trabalho.

O capítulo 2 descreve os conceitos teóricos básicos para o desenvolvimento do sistema de controle de variador de velocidade usando IoT.

No capítulo 3 é mostrado o circuito de controle.

O capítulo 4 mostra os resultados experimentais que indicam a efetividade do sistema de controle IoT.

Finalmente, conclusões e trabalhos futuros são indicados.

---

## 2 FUNDAMENTAÇÃO TEÓRICA

---

### 2.1 TEORIA DOS FILTROS ANALÓGICOS

Filtros analógicos tratam-se de circuitos lineares com objetivo de atenuar certas frequências, enquanto permitem a passagem de outras. A seguir, alguns conceitos fundamentais para o seu entendimento serão explicados.

#### 2.1.1 Decibel (dB)

O conceito de decibel (dB), foi criado para expressar a relação logarítmica entre dois sinais de potência ou intensidade diferentes.

Por intermédio da empresa Bell Labs, foi definida uma unidade que pudesse mensurar as perdas de transmissão de sinais. Foi assim, criado o Bel ( $\beta$ ), como homenagem à Alexandre Graham Bell, por suas contribuições na área de telefonia (CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU, 2013).

O Bel é utilizado para mensurar a relação de um sinal de saída pelo seu sinal de entrada.

$$G_p(B) = \log\left(\frac{P_o}{P_i}\right) \quad (1)$$

O decibel é uma unidade de medida adimensional comparativa, utilizada na teoria de telecomunicações, processamento de sinais e controle. Ele representa a décima parte da unidade Bel, e é constantemente utilizado com mais frequência que este, uma vez que o Bel é uma unidade grande para propósitos práticos (ROBERT L. BOYLESTAD; LOUIS NASHELSKY, 2013).

$$G_p(dB) = 10 \log\left(\frac{P_o}{P_i}\right) \quad (2)$$

Há uma segunda forma de representar o ganho, mais comum em análises de filtros analógicos (ANA LUÍSA FUNARI PIMENTA, 2018) . Considerando que para um ganho de saída  $P_o$ , há uma tensão  $V_o$  associada, e para uma potência de entrada  $P_i$ , há  $V_i$ , tem-se:

$$P_o = \frac{V_o^2}{R_o} \quad (3)$$

---


$$P_i = \frac{V_i^2}{R_i} \quad (4)$$

Considerando  $R_1=R_2$ , a relação entre as tensões fica:

$$G_v(dB) = 10 \log \left( \frac{V_o}{V_i} \right)^2 = 20 \log \left( \frac{V_o}{V_i} \right) = 20 \log(G_v) \quad (5)$$

Convertendo o ganho de tensão em decibel para o ganho linear de tensão, tem-se:

$$G_v = 10^{\frac{G_v(dB)}{20}} \quad (6)$$

Tabela 1 - Ganho de potência linear, em Bel e em decibel.

| Ganho em Po/Pi | Ganho em Bel ( $\beta$ ) | Ganho em decibel (dB) |
|----------------|--------------------------|-----------------------|
| 0,5            | 0,3010                   | -3,0103               |
| 0,707          | 0,1506                   | -1,5058               |
| 1              | 0,0000                   | 0,0000                |
| 10             | 1,0000                   | 10,0000               |
| 20             | 1,3010                   | 13,0103               |
| 30             | 1,4771                   | 14,7712               |
| 40             | 1,6021                   | 16,0206               |
| 50             | 1,6990                   | 16,9897               |
| 60             | 1,7782                   | 17,7815               |
| 70             | 1,8451                   | 18,4510               |
| 80             | 1,9031                   | 19,0309               |
| 90             | 1,9542                   | 19,5424               |
| 100            | 2,0000                   | 20,0000               |

Fonte: Adaptado de (ROBERT L. BOYLESTAD; LOUIS NASHESKY, 2013).

Tabela 2 - Ganho de tensão linear, em Bel e em decibel.

| Ganho em $V_o/V_i$ | Ganho em Bel ( $\beta$ ) | Ganho em decibel (dB) |
|--------------------|--------------------------|-----------------------|
| 0,5                | 0,6021                   | -6,0206               |
| 0,707              | 0,3012                   | -3,0116               |
| 1                  | 0,0000                   | 0,0000                |
| 10                 | 2,0000                   | 20,0000               |
| 20                 | 2,6021                   | 26,0206               |
| 30                 | 2,9542                   | 29,5424               |
| 40                 | 3,2041                   | 32,0412               |
| 50                 | 3,3979                   | 33,9794               |
| 60                 | 3,5563                   | 35,5630               |
| 70                 | 3,6902                   | 36,9020               |
| 80                 | 3,8062                   | 38,0618               |
| 90                 | 3,9085                   | 39,0849               |
| 100                | 4,0000                   | 40,0000               |

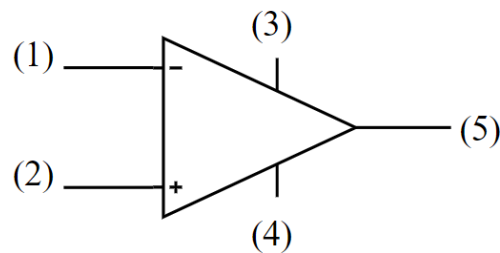
Fonte: Adaptado de (ROBERT L. BOYLESTAD; LOUIS NASHELSKY, 2013).

Ao dobrar o ganho de potência (ou de tensão) em um amplificador, o ganho em Bel ou em decibel não dobra o seu valor, isso porque essas unidades adimensionais obedecem a escala logarítmica, e só dobram o seu valor após a potência ser amplificada em 10 vezes, como demonstrado nas tabelas 1 e 2 acima.

### 2.1.2 Amplificadores operacionais

O Amplificador Operacional (AmpOp) é um componente eletrônico de amplificação e processamento de sinais, composto por uma entrada diferencial, com ganho alto, impedância de entrada alta e baixa impedância de saída (ROBERT L. BOYLESTAD; LOUIS NASHELSKY, 2013). Suas aplicações estão presentes nos sistemas de controle industrial, na instrumentação industrial, nas telecomunicações, nos equipamentos de áudio e nos sistemas de aquisição de dados (ANTONIO PERTENCE JR., 2014).

---

 Figura 2 - AmpOp elementar.


Fonte: Adaptado de (ANTONIO PERTENCE JR., 2014) .

A descrição dos pinos do AmpOp dado pela figura 3 é:

- (1) – Entrada inversora ( $V_-$ );
- (2) – Entrada não inversora ( $V_+$ );
- (3) – Alimentação negativa (-3V a -18V);
- (4) – Alimentação positiva (+3V a +18V);
- (5) – Saída.

Cada entrada do AmpOp, resulta em uma saída de mesma polaridade, podendo ser de mesma fase (+), ou de fase oposta (-). O sinal da tensão de saída é proporcional à diferença de tensões de sua entrada, dado pela expressão:

$$V_o = A_v \cdot (V_+ - V_-) \quad (7)$$

Sendo  $A_v$  o ganho linear de tensão.

Esse ganho de tensão dado por  $A_v$ , pode ser expressado em decibéis, por meio da equação (5), explicada anteriormente. Em decibéis, a representação gráfica é facilitada em uma ampla gama de frequências.

Dentre as condições fundamentais para um adequado funcionamento de um amplificador operacional (ANTONIO PERTENCE JR., 2014; WENDLING, 2010), destacam-se:

- Ganho de tensão alto, idealmente até o infinito;
- Resposta em frequência com uma largura de faixa ampla;
- Baixa sensibilidade à temperatura, conhecido como “DRIFT”.

Um AmpOp pode operado de três modos

- Sem realimentação (ou malha aberta);
- Com realimentação positiva;
- Com realimentação negativa.

---

Dentre esses três modos de operação, o usado para este projeto será o modo com realimentação negativa, sendo esse o método de malha fechada mais importante em circuitos com AmpOp (ANTONIO PERTENCE JR., 2014).

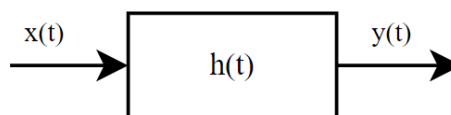
### 2.1.3 Resposta Em Frequência

Considerando um sinal de entrada de amplitude e fase constantes, com a frequência variável, a resposta em frequência visa estabelecer o comportamento do sinal de saída, com amplitude e fase variável, em função da frequência. Trata-se da análise do comportamento do sistema em uma determinada faixa de frequência (ANA LUÍSA FUNARI PIMENTA, 2018). É a resposta em estado estacionário da planta a uma entrada de sinal sinusoidal com frequência variável (CORDERO et al., 2022).

Resposta  $y(t)$  estacionária da planta a uma entrada senoidal  $x(t)$  de frequência variável. Em sistema LIT (Sistema Linear Invariante no Tempo), depende da resposta ao impulso  $h(t)$  (DANIEL CRUVINEL CORRAL; GUILHERME AFONSO MAZANTI, 2011). Isso pode ser representado por meio da equação:

$$y(t) = h(t) * x(t) \quad (8)$$

Figura 3 - Diagrama de bloco elementar de um LIT



Fonte: Próprio autor.

Com auxílio da transformada de Laplace, uma função de transferência em um sistema LIT pode ser representado no domínio  $s$ , considerando todas as condições iniciais nulas.

$$H(s) = \frac{Y(s)}{X(s)} \quad (9)$$

A descrição de uma função de transferência se dá por uma expressão polinomial, em seu numerador e denominador. As raízes que compõem o seu numerador são denominados zeros, e as raízes de seu denominador são denominados polos (ANA LUÍSA FUNARI PIMENTA, 2018). Assim, pode-se representar a equação (12) como a razão de 2 polinômios:

---


$$H(s) = \frac{a_m \cdot s^m + a_{m-1} \cdot s^{m-1} + \dots + a_0}{b_n \cdot s^n + b_{n-1} \cdot s^{n-1} + \dots + b_0} \quad (10)$$

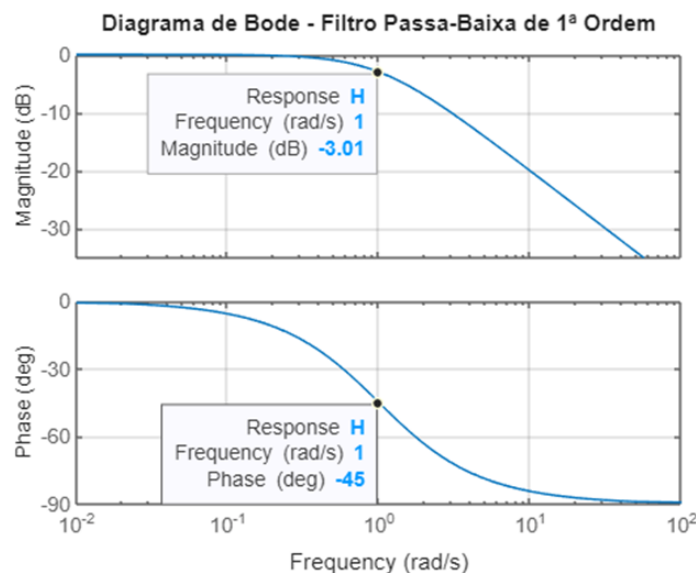
O grau  $n$  do denominador representa a ordem do filtro, enquanto  $m$  representa o grau do numerador, onde  $n \geq m$  para um filtro estável (ANA LUÍSA FUNARI PIMENTA, 2018)

#### 2.1.4 Diagrama de Bode

Os diagramas de Bode receberam esse nome em homenagem ao engenheiro Hendrik W. Bode, da empresa Bell Telephone Laboratories, pelo seu trabalho pioneiro na teoria de controle e sistemas de telecomunicações na década de 1940.

Uma forma prática e sistemática de representar graficamente a resposta em frequência de uma função de transferência é através do diagrama de Bode, que consiste em dois gráficos semilogarítmicos da magnitude e da fase, ambos em função da frequência (ANA LUÍSA FUNARI PIMENTA, 2018; CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU, 2013).

Figura 4 - Diagrama de Bode de um polo simples.



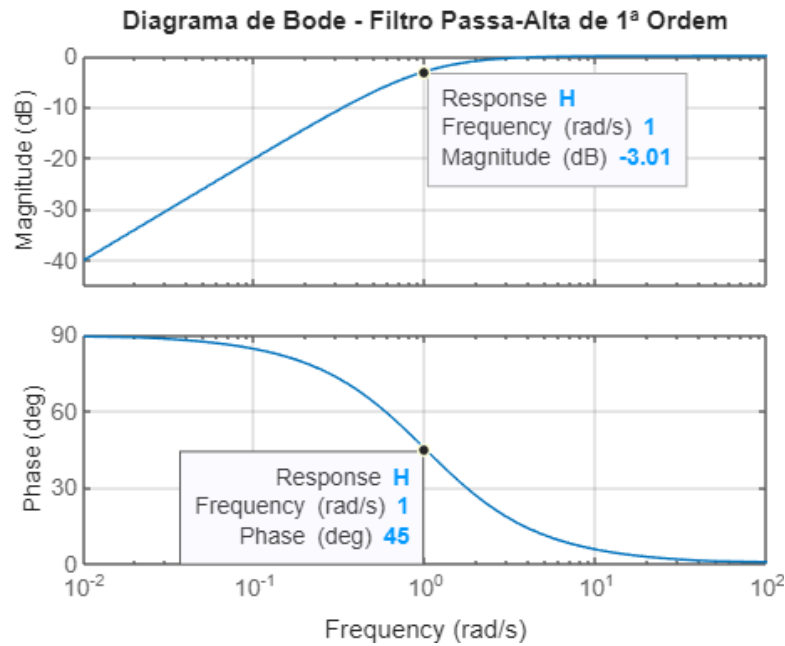
Fonte: Próprio autor.

Para um polo simples, após a frequência de corte, a atenuação do sinal é de  $-20\text{dB}/\text{déc}$  (a cada década de frequência, o ganho atenua em 20dB), enquanto na fase, em baixas frequências tem um ângulo de  $0^\circ$ , enquanto em altas frequências tem um ângulo de  $90^\circ$  (CHARLES K.

ALEXANDER; MATTHEW N. O. SADIKU, 2013). Este é o efeito do integrador, representado por:

$$H(\omega) = \frac{1}{1+j\omega RC} \quad (11)$$

Figura 5 - Diagrama de Bode de um zero simples.



Fonte: Adaptado de (CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU, 2013)

O efeito do zero simples faz com que a inclinação do gráfico do ganho, antes da frequência de corte, seja de  $+20\text{dB/déc}$ , enquanto no gráfico da fase, em baixas frequências, a fase é de  $90^\circ$ , enquanto para altas frequências, a fase é de  $0^\circ$ .

$$H(\omega) = \frac{j\omega L}{R+j\omega L} \quad (12)$$

### 2.1.5 Filtros Analógicos

Filtros eletrônicos são circuitos eletrônicos lineares desenvolvidos com o objetivo de selecionar uma determinada largura de frequência de um sinal, enquanto atenua as demais, apresentando a resposta em frequência desejada. Esse tipo de aplicação é muito utilizado na eletrônica, com a seleção de estações de frequência para aparelhos de rádio e TV, e na telefonia para separar uma frequência de transmissão de outra (CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU, 2013).

---

Em relação a tecnologia de um filtro eletrônico, ele pode ser categorizado de três formas:

- Filtros passivos;
- Filtros ativos;
- Filtros digitais.

O primeiro e o segundo tipos de filtro são classificados como filtros analógicos.

Filtros passivos são circuitos formados apenas por elementos passivos, não geradores de energia, como resistores, capacitores e indutores. Os filtros ativos, além da presença dos componentes passivos, possuem componentes geradores de energia, como os amplificadores operacionais e transistores (ANA LUÍSA FUNARI PIMENTA, 2018).

Uma classificação que remete à função dos filtros eletrônicos apresenta quatro tipos de filtros, com respostas em frequência distintas:

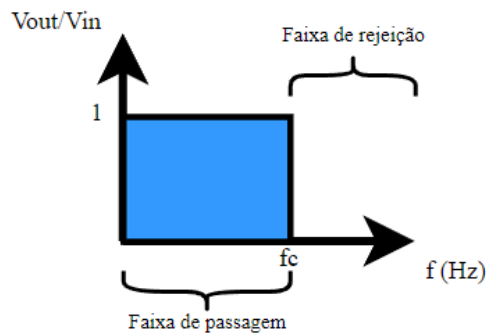
- Filtro Passa Baixas (PB);
- Filtro Passa Alta (PA);
- Filtro Passa Faixa (PF);
- Filtro Rejeita Faixa (RF).

#### **2.1.5.1 Filtro Passa Baixa (PB) de 1ª ordem**

O filtro PB é responsável por atenuar os sinais de alta frequência, acima da frequência de corte  $f_c$ , enquanto permite baixas frequências, abaixo da frequência de corte. A ordem de um filtro, no geral, é classificada com base no grau do polinômio do seu denominador (ANA LUÍSA FUNARI PIMENTA, 2018).

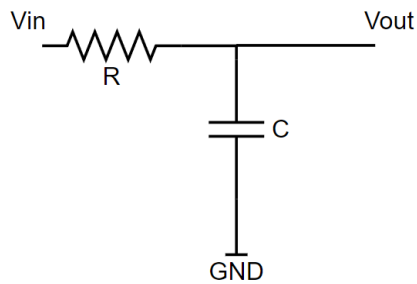
Em um filtro PB ideal, antes da frequência de corte  $f_c$ , há um ganho de tensão que pode ser igual a 1, com uma tensão de saída com magnitude próxima a tensão de entrada. Já para sinais após a  $f_c$ , o ganho de tensão é nulo, com tensão de saída igual a zero. Além disso, em um filtro ideal, não há sinais provenientes da faixa de rejeição, pois a faixa de transição está ausente, aspecto este inexistente no projeto de um filtro real (ANA LUÍSA FUNARI PIMENTA, 2018; DANIEL CRUVINEL CORRAL; GUILHERME AFONSO MAZANTI, 2011).

Figura 6 - Gráfico com a resposta em frequência do ganho de um filtro PB ideal.



Fonte: Próprio autor.

Figura 7 - Circuito RC de um filtro passa baixas passivo de 1ª ordem.



Fonte: Próprio autor.

A fim de se calcular um filtro passa baixas de 1ª ordem qualquer, a sua função de transferência é normalizada em relação a  $fc$ , utilizando a variável  $s = \frac{1}{\omega c}$ . (DANIEL CRUVINEL CORRAL; GUILHERME AFONSO MAZANTI, 2011). Logo, a sua função de transferência, com um polo real fica:

$$H(s) = \frac{\frac{1}{Cs}}{R + \frac{1}{Cs}} = \frac{1}{RCs + 1} = \frac{1}{\tau s + 1} \quad (13)$$

A frequência de corte pode ser calculada por meio de:

$$fc = \frac{1}{2\pi RC} \quad (14)$$

### 2.1.5.2 Filtro Passa Baixa (PB) de 2ª ordem com topologia Sallen-Key

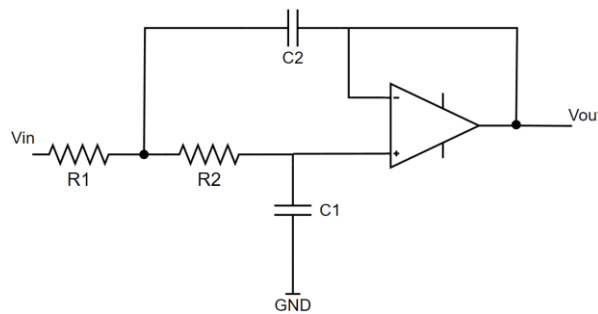
Uma topologia comum para filtros PB de 2ª ordem é a topologia Sallen-Key, que utiliza resistores e capacitores, além de um amplificador operacional (DANIEL CRUVINEL CORRAL;

GUILHERME AFONSO MAZANTI, 2011; TEXAS INSTRUMENTS, 2008). A configuração de sua função de transferência fica:

$$H(s) = \frac{A_0}{(1+a_1s+b_1s^2)(1+a_2s+b_2s^2)} \cdots \frac{A_0}{1+a_1s+b_1s^2} \quad (15)$$

Sendo  $A_0$ , o ganho da função,  $a_1$  e  $b_1$  os coeficientes reais do filtro. Essa função de transferência apresenta no denominador uma série de polinômios de segunda ordem, em cascata.

Figura 8 - Circuito de um filtro passa baixas ativo de 2ª ordem com ganho unitário, de Sallen-Key.



Fonte: Adaptado de (TEXAS INSTRUMENTS, 2008).

A função de transferência que expressa esse circuito, fica:

$$H(s) = \frac{1}{1 + \omega_c C_1 (R_1 + R_2) s + \omega_c^2 R_1 R_2 C_1 C_2 s^2} \quad (16)$$

Fixando-se os valores dos capacitores, os componentes do circuito do filtro podem ser calculados, como:

$$R_{1,2} = \frac{a_1 C_2 \mp \sqrt{a_1^2 C_2^2 - 4 b_1 C_1 C_2}}{4 \pi f_c C_1 C_2} \quad (17)$$

$$C_2 \geq C_1 \cdot \left( 4 \cdot \frac{b_1}{a_1^2} \right) \quad (18)$$

Considerando o ganho unitário  $A_0 = 1$ , os coeficientes reais podem ser calculados como (DANIEL CRUVINEL CORRAL; GUILHERME AFONSO MAZANTI, 2011; TEXAS INSTRUMENTS, 2008):

$$a_1 = \omega_c C_1 (R_1 + R_2) \quad (19)$$

$$b_1 = \omega_c^2 R_1 R_2 C_1 C_2 \quad (20)$$

### 2.1.5.3 Resposta em frequência de filtros ativos reais

Em comparação a filtros ideais, os filtros reais (em qualquer ordem), apresentam alguns problemas (TEXAS INSTRUMENTS, 2008):

- A sua faixa de transição, a partir da atenuação de -3dB, não acontece de forma abrupta, mas sim de maneira gradual deslocando a atenuação real.
- A resposta da fase não é linear, isso acrescenta sinais de distorção significativos.

O ganho e a fase da resposta em frequência de um filtro real podem ser otimizados da seguinte forma:

- Garantir que as faixas de passagem e rejeição sejam as mais planas possíveis
- Garantir que a faixa de transição seja a menor possível
- Uma resposta da fase o mais linear possível

Para uma implementação prática dos filtros ativos reais, uma outra classificação é utilizada, considerando-se o formato da resposta em frequência, com tipos de resposta distintas, e com determinações específicas, sendo elas:

- O filtro de Butterworth, que possui as faixas passagem e de rejeição planas, enquanto sua faixa de transição é suave.
- O filtro de Tschhebyscheff, dividida em dois tipos, com ambos possuindo uma faixa de transição moderada, enquanto o primeiro tem sua faixa de rejeição plana (Tschhebyscheff 1), e o outro tem sua faixa de transição plana (Tschhebyscheff 2).
- O filtro de Bessel, que possui as faixas de transição e rejeição planas, enquanto a sua faixa de transição é suave, semelhante ao de Butterworth.

Considerando a topologia de Sallen-Key apresentada anteriormente, os coeficientes reais de cada tipo de resposta são distintos, sendo eles:

Tabela 3 - Coeficientes de filtros de 2ª ordem.

| 2ª ORDEM | BESSEL | BUTTERWORTH | TSCHEBYSCHIEFF |
|----------|--------|-------------|----------------|
| $a_1$    | 1,3617 | 1,4142      | 1,065          |
| $b_1$    | 0,618  | 1           | 1,9305         |

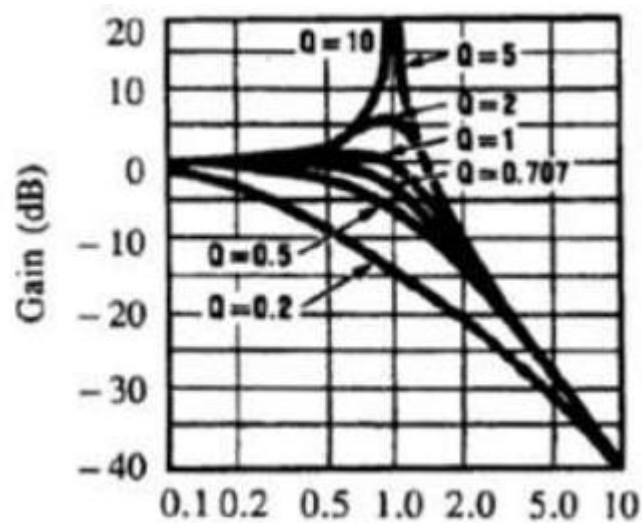
|     |      |      |     |
|-----|------|------|-----|
| $Q$ | 0,58 | 0,71 | 1,3 |
|-----|------|------|-----|

Fonte: (TEXAS INSTRUMENTS, 2008).

O fator de qualidade  $Q$  trata-se de um importante parâmetro de projeto, representando a qualidade do polo, pode ser calculado pela expressão:

$$Q = \frac{\sqrt{b_1}}{a_1} \quad (21)$$

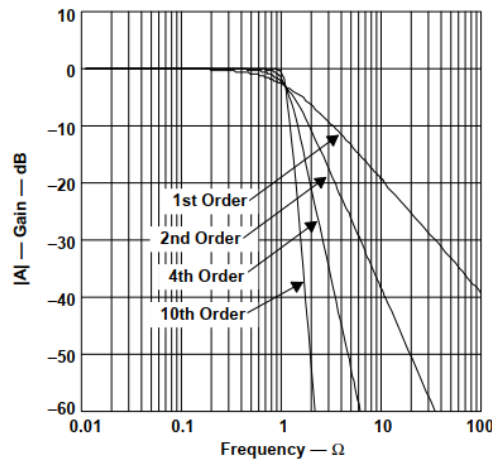
Figura 9 - Respostas de filtros de 2ª ordem com diferentes valores de  $Q$ .



Fonte: (ANA LUÍSA FUNARI PIMENTA, 2018)

Outro parâmetro importante, é a ordem do filtro, sendo um número inteiro que implica diretamente no decaimento de sua faixa de transição. Filtros de ordem elevada apresentam maior taxa de atenuação/ganho, se aproximando da situação de um filtro ideal, com faixa de transição ausente.

Figura 10 - Resposta em frequência da magnitude de filtros Butterworth, de 1ª, 2ª, 4ª e 10ª ordem.



Fonte: (TEXAS INSTRUMENTS, 2008)

## 2.2 ESP-32

O ESP32 é uma série de microcontroladores SOC (System On Chip – Sistema em um Chip), desenvolvidos pela empresa chinesa Espressif Systems, sediada em Xangai. É composto por um módulo Wi-fi com conectividade 802.11b/g/n e Bluetooth de modo duplo integrados em sua placa de desenvolvimento, imprescindível para aplicações em IoT (MAREK BABIUCH; PETR FOLTÝNEK; PAVEL SMUTNÝ, 2019).

Este microcontrolador, da segunda geração de microcontroladores com soluções IoT, fabricados pela Espressif, estabelece-se como sucessor direto do microcontrolador ESP8266. Esta série de microcontroladores compõe um microprocessador Xtensa LX6, fabricado pela empresa Tensilica, com arquitetura de 32 bits, em duas versões: de 1 ou 2 núcleos (single-core ou dual-core). A velocidade de clock é de até 240 MHz, com 520 KB de SRAM (“Static Random Access Memory” – Memória de Acesso Aleatório Estático) (DOKIC, 2020).

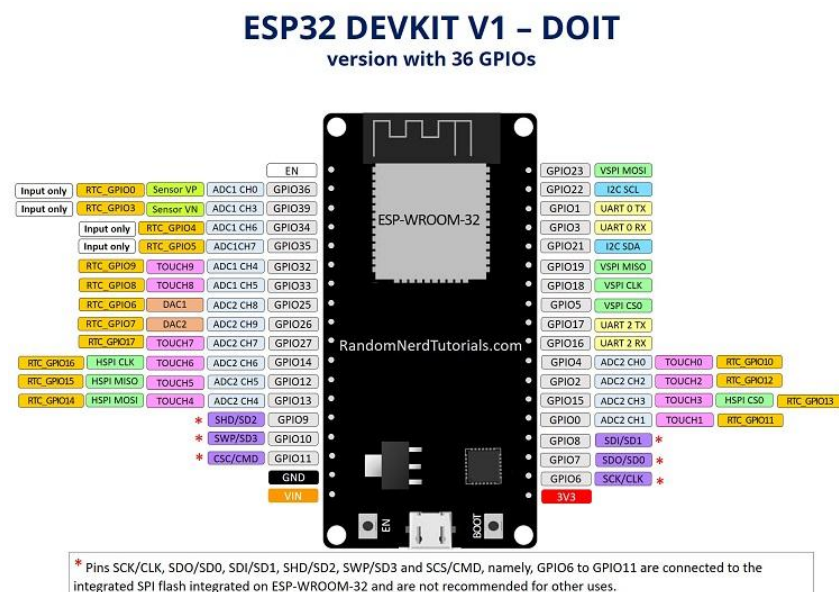
Em comparação ao ESP 8266, o ESP32 oferece maior poder de processamento, um módulo Wi-fi mais rápido e com maior alcance, graças a uma antena presente em sua placa. Este microcontrolador possui também suporte para duas versões Bluetooth, versão 4.2 e LE (Low Energy), enquanto a versão anterior não suporta a tecnologia Bluetooth. Ainda, o ESP32 conta com mais canais de entrada e saída (GPIO), sendo a sua única desvantagem em relação ao seu antecessor, o seu custo mais elevado, embora as suas funcionalidades adicionais possam justificar o investimento mais elevado (RIOS, 2019).

Tabela 4 – Especificações do ESP8266 e ESP32

|                              | ESP8266                 | ESP32                         |
|------------------------------|-------------------------|-------------------------------|
| Wifi 802.11b/g/n             | 2.4GHz, 72.2 Mbps       | 2.4GHz, 150 Mbps              |
| Bluetooth                    | Não suporta             | Bluetooth v4.2 e Bluetooth LE |
| CPU                          | Xtensa L106 de 1 núcleo | Xtensa LX6 de 1 ou 2 núcleos  |
| Arquitetura                  | 32 bits                 | 32 bits                       |
| Frequência do Clock          | 80 MHz                  | 160 a 240 MHz                 |
| Memória RAM                  | 50 KB RAM               | 520 KB SRAM                   |
| Memória ROM                  | Não suporta             | 448 KB ROM                    |
| Memória Flash                | 512 KB                  | 4 MB                          |
| GPIO                         | 17 canais               | 36 canais                     |
| Conversor AD                 | 1 canal de 10 bits      | 18 canais de 12 bits          |
| Conversor DA                 | Não suporta             | 2 canais de 8 bits            |
| SPI/I2C/I2S/UART<br>(Serial) | 2/1/2/2 canais          | 4/2/2/2 canais                |
| PWM                          | 8 canais                | 16 canais                     |
| Sensor de toque capacitivo   | Não suporta             | 10 canais                     |

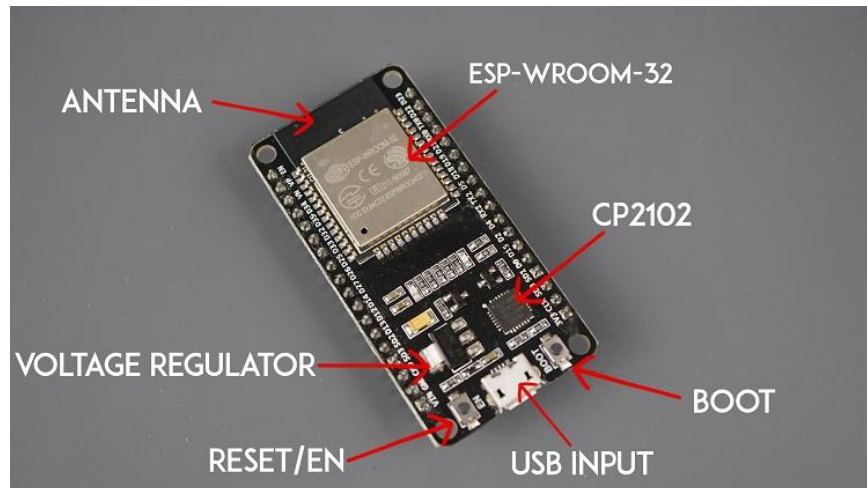
Fonte: (ESPRESSIF SYSTEMS, 2023, 2024; RIOS, 2019)

Figura 11 - Pinagem do microcontrolador ESP32.



Fonte: (RANDOM NERD TUTORIALS, 2024)

Figura 12 - Microcontrolador ESP32



Fonte: (RANDOM NERD TUTORIALS, 2024)

Sobre os principais periféricos do ESP32, estes possuem um conector micro USB para alimentação (5V) e comunicação de dados com o computador/PC. Também está presente em sua placa, um regulador de tensão e uma interface UART-USB pelo chip CP2102, garantindo melhor compatibilidade de comunicação com o computador. Os botões BOOT e RESET/EN servem, respectivamente, para enviar e reiniciar o código ao ESP32.

A programação dos microcontroladores ESP8266 e ESP32 pode ser escrita na linguagem C++, de forma acessível, utilizando o software de desenvolvimento gratuito Arduino IDE (Integrated Development Environment – Ambiente de Desenvolvimento Integrado), da empresa Arduino (ARDUINO, [s.d.]).

Desde o seu lançamento, em 06 de setembro de 2016, a popularidade dessa série aumentou de forma significativa, dado pelo seu valor competitivo de mercado, aliado a projeção de popularidade associada à projetos em IoT.

Figura 13 - Gráfico de interesse ao longo do tempo na pesquisa do termo "ESP32" no mecanismo de pesquisa Google, no mundo. (período de 06.09.2016 a 15.10.2024).



---

Fonte: (GOOGLE TRENDS, [s.d.])

### 2.2.1 Rede Wi-fi

O padrão IEEE 802.11, também conhecido como rede Wi-fi, é um protocolo de comunicação de dados sem fio, que define uma série de padrões para transmissão e codificação de dados. Seu desenvolvimento foi pensado como uma alternativa ao protocolo Ethernet, com altas taxas de dados e baixo custo relativo. Seu uso é comum em escritórios, comércios, residências e locais públicos, adequado para a conexão com dispositivos móveis (SANTOS et al., 2019).

Desde o seu lançamento oficial, em 1997, pelo comitê de padronização do IEEE, novas versões do padrão 802.11 foram aprovadas, onde, o último padrão lançado oficialmente é o 802.11be, também conhecido como Wi-fi 7, com taxas de transmissão de até 8,64 Gbps, atuando nas bandas de 2,4, 5 e 6 GHz (EHUD RESHEF; CARLOS CORDEIRO, 2022).

Os protocolos compatíveis com os microcontroladores ESP 8266 e ESP32, por exemplo, para transmissão de dados sem fio, incorporam 3 padrões lançados pela IEEE: o padrão 802.11b/g/n. No ESP32, essa comunicação opera na frequência de 2.4GHz, em uma taxa de transferência de dados de 150Mbps. No microcontrolador ESP8266, a frequência de operação é idêntica, com uma taxa de transferência de dados de 72Mbps (ESPRESSIF SYSTEMS, 2023, 2024).

## 2.3 INTERNET DAS COISAS (IoT)

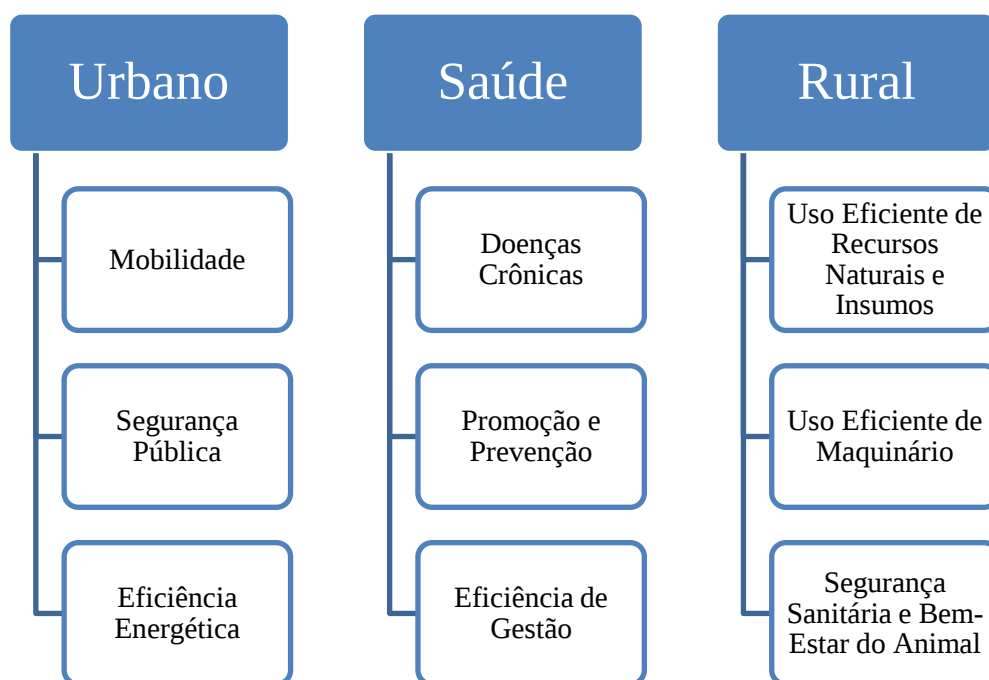
O termo Internet das Coisas (IoT – “*Internet of Things*”), diz respeito a incorporação de dispositivos elétricos ou eletrônicos, com capacidade de processamento, a um sistema de comunicação conectado via internet, permitindo a transferência, coleta e troca de dados entre diferentes dispositivos (ELOI; FRANÇA; CAMPOS, 2023).

Trata-se de uma expansão da internet, com a integração do mundo físico dos dispositivos eletrodomésticos ou industriais, com o mundo virtual, onde diferentes tipos de dispositivos interagem entre si de forma remota. Isso gera oportunidades de pesquisa no âmbito acadêmico, e aplicações inteligentes no ramo industrial (SANTOS et al., 2019).

De acordo com o decreto intitulado “Plano Nacional de Internet das Coisas” (PRESIDÊNCIA DA REPÚBLICA, 2019), publicado com o objetivo de alavancar os projetos nesta área no país, a Internet das Coisas pode também ser definida como a infraestrutura que integra dispositivos de conexão física a dispositivos de tecnologia da informação, com interoperabilidade.

Além do decreto, no Brasil, o BNDES (Banco Nacional de Desenvolvimento Econômico e Social), financia projetos e soluções tecnológicas e inovadoras em IoT por meio de um programa chamado “BNDES Pilotos IoT”, em parceria com o MCTIC (Ministério da Ciência, Tecnologia, Inovações e Comunicações) e conduzido pelo consórcio McKinsey (BNDES, 2017, 2018). Nesse programa, cada projeto IoT é financiado com o valor mínimo de R\$ 1 milhão. Podem ser apoiadas Instituições Tecnológicas (IT) e Instituições de Apoio (IA), cujo apoio acontece principalmente em 3 setores prioritários: setor urbano, setor de saúde e setor rural.

Figura 14 - BNDES Pilotos IoT – 3 áreas priorizadas.



Fonte: Adaptado de (BNDES, 2018)

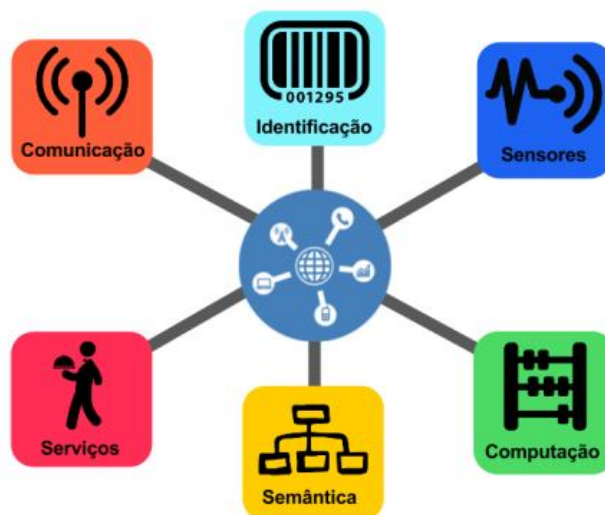
Segundo um relatório publicado pelo consórcio McKinsey, a IOT tem aplicações significativas nos setores de saúde, transporte e manufatura, e até 2030, estima-se que gere 5,5 trilhões de dólares e 12,6 trilhões de dólares em valor. Entretanto, alguns desafios são colocados em pauta, como a falta de padronização, a complexidade de desenvolvimento de dispositivos

conectados à rede e a escassez de semicondutores, agravado pela pandemia de covid-19 (MCKINSEY, 2023).

Hoje, há uma variedade de aparelhos com capacidade computacional conectados à internet, desde computadores e celulares a automóveis e eletrodomésticos (SANTOS et al., 2019). Dada essa pluralidade, estima-se que até 2025, 27 bilhões de aparelhos estarão conectados (FORBES, 2022).

Em relação a estrutura básica de projetos IOT, segundo (SANTOS et al., 2019), uma aplicação em Internet das Coisas pode ser descrita em 6 blocos básicos, sendo eles: Comunicação, Identificação, Sensores, Serviços, Semântica e Computação.

Figura 15 - Blocos básicos de IoT.



Fonte: (SANTOS et al., 2019)

- Comunicação: É fundamental em qualquer aplicação IOT. Seja uma comunicação via Wi-fi ou Bluetooth, o equipamento deve ser capaz de se comunicar com outros dispositivos.
- Identificação: Consiste em identificar os dispositivos conectados a uma rede, por meio do seu endereço IP.
- Sensores/Atuadores: Os sensores são responsáveis por receber um sinal de entrada do ambiente, e o encaminha para um dispositivo (microcontrolador, por exemplo), onde será processado. Os atuadores são responsáveis por atuar no ambiente, por meio de um sinal de saída, conforme o sinal processado.

- 
- **Serviços:** São serviços inteligentes na IOT, que funcionam rotineiramente sem serem notados pelo usuário. Destacam-se aqui: Serviços de Identificação, Serviços de Agregação de Dados, Serviços de Colaboração e Inteligência e Serviços de Ubiquidade.
  - **Semântica:** Neste bloco, a função é agregar conhecimento, melhorando os serviços em IOT.
  - **Computação:** É a parcela responsável por processar os dados recebidos por sensores ou outros dispositivos conectados.

## 2.4 HTML

A primeira versão do HTML (*“HyperText Markup Language”* – Linguagem de Marcação de Hipertexto) foi criada pelo físico e professor britânico do MIT, Timothy John Berners-Lee, enquanto atuava como pesquisador na Seção de Computação da Organização Europeia para a Pesquisa Nuclear (CERN). Ele desenvolveu um método que permitia a um grupo de pesquisadores da época compartilhar, de forma eletrônica, textos de suas pesquisas (Mauricio Samy Silva, 2019).

Ao longo do desenvolvimento da linguagem HTML, em 2008, o HTML5 foi publicado pelo W3C (*“World Wide Web Consortium”* – Consórcio World Wide Web), influenciado pelo trabalho conjunto realizado pelas empresas Apple, Mozilla e Opera, conhecido como WHATWG (Web Hypertext Application Technology Working Group). Esse grupo surgiu como uma contraposição aos padrões exigidos pelo W3C em 2002, com a criação do XHTML 2.0 (Fábio Flatschart, 2011). Em 2014, o HTML5 foi oficialmente recomendado pela W3C (W3C, 2014), vigente até os dias atuais, introduz funcionalidades às especificações CSS3 e Javascript, garantindo páginas web mais dinâmicas e interativas (MENEZES, 2013).

A linguagem HTML é uma linguagem de marcação para criação e desenvolvimento de páginas web. Ela é responsável por estruturar uma página, permitindo definir cabeçalhos, parágrafos, criar blocos, inserir imagens, inserir links e outras aplicações (INDIRA NIGHT, 2018).

Um arquivo HTML é geralmente referenciado como *“index.html”*. A figura 16 apresenta a estrutura básica desse arquivo.

Figura 16 - Exemplo básico da estrutura de um arquivo HTML5.

```
<> index.html > ...
1  <!DOCTYPE html> <!-- Declaração de um arquivo HTML5 -->
2  <html lang="en">
3  <head> <!-- Encapsula todas as informações da página, sem mostrar o seu conteúdo diretamente -->
4      <meta charset="UTF-8"> <!-- Informa ao navegador o tipo de codificação de caracteres -->
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <link rel="stylesheet" href="style.css"> <!-- Importa o arquivo CSS para este -->
7      <script src="js/scripts.js" defer></script>
8      <title>Document</title> <!-- Título que aparece na aba do navegador -->
9  </head>
10 <body> <!-- Armazena a escrita da página web -->
11     <h1>TÍTULO TESTE</h1>
12     <p>Isto é um parágrafo</p>
13     <div id="saidaJS"></div> <!-- ID referente a saída do JS -->
14 </body>
15 </html>
```

Fonte: Próprio autor.

## 2.4.1 CSS E JAVASCRIPT

A linguagem HTML sozinha não consegue implementar aplicações em páginas web, por definir conteúdos, e não ser uma linguagem de programação, e sim, de marcação. Assim, é necessário o uso das linguagens CSS e Javascript.

O CSS (“Cascading Style Sheets” – Folhas de Estilo em Cascata) é associado à estilização dos elementos de marcação do HTML, modificando suas características físicas (cores, tamanho, fonte), ou mesmo as suas posições na página web (INDIRA NIGHT, 2018).

Um arquivo em CSS geralmente é escrito como “style.css”. Sua escrita é dividida em vários seletores, separados por “{}”. Há a possibilidade de escrever o CSS diretamente no navegador, mas trata-se de uma boa prática associar um arquivo separado para isso (INDIRA NIGHT, 2018).

---

Figura 17 - Exemplo básico de um arquivo CSS3.

```
# style.css > ...
1  *{ /* Seletor Universal */
2  |   font-family: Arial, Helvetica, sans-serif; /* Lista as fontes */
3  | }
4  h1{
5  |   color: ■red; /* Muda a cor do texto em h1 */
6  |   border: ■red solid 1px; /* cria uma borda em volta do texto h1*/
7  | }
8  p{
9  |   color: ■blue; /* Muda a cor do texto e p*/
10 | }
11 #saidaJS{
12 |   font-weight: bold;
13 | }
```

Fonte: Próprio autor.

Javascript é uma linguagem de programação associada a objetos, desenvolvida pela empresa Netscape, em parceria com a Sun Microsystems, implementada em meados de 1996. Sua programação foi formulada para ser implementada no lado do cliente (“front-end”), com suas funcionalidades dependentes do navegador do usuário (MENEZES, 2013).

Associado ao HTML, o Javascript permite que os elementos funcionem de maneira interativa, de forma que eles possam ser entendidos pelo computador, através das instruções do escritas no código (INDIRA NIGHT, 2018).

---

 Figura 18 - Estrutura básica de um arquivo Javascript.

```

js > JS scripts.js > ...
1  // Declaração de variáveis
2  var numero = 1;
3  var resultado = "";
4
5  // Início do loop
6  while(numero < 10){
7  resultado += numero + " "; // adiciona um número e um espaço
8  numero++; // incrementa o número
9  }
10 // Exibe o resultado na página
11 document.getElementById("saidaJS").innerHTML = resultado;
  
```

Fonte: Próprio autor.

Figura 19 - Página web simples criada.

**TÍTULO TESTE**

Isto é um parágrafo

**1 2 3 4 5 6 7 8 9**

Fonte: Próprio autor.

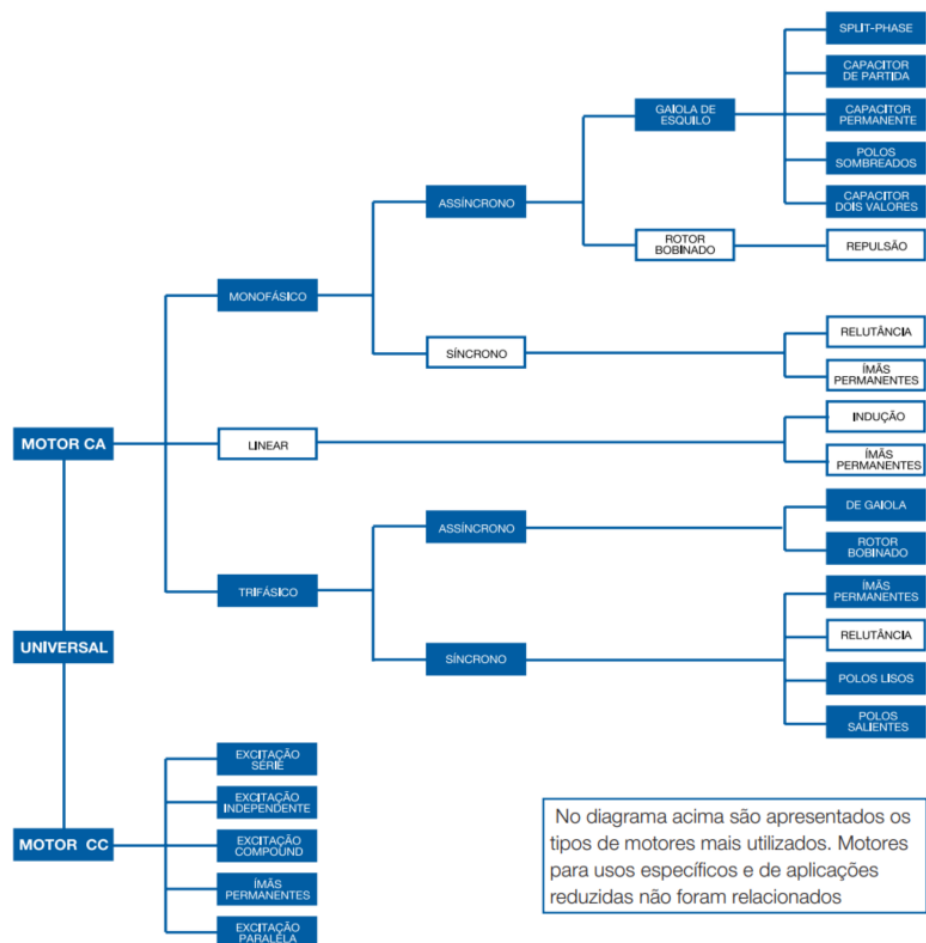
## 2.5 MOTOR DE INDUÇÃO

As ideias atribuídas a criação do motor de indução remetem as figuras de Nicola Tesla, Galileu Ferraris e Michael Von Dolivo-Dobrovolski. Em 1885, Galileu Ferraris elaborou os princípios básicos associados ao campo girante de um motor de indução, porém, sua criação apresentava um rendimento médio de 50%. Tesla, em 1887, efetivou um pequeno protótipo de um motor de indução bifásico, mas seu rendimento ainda era precário. Dado o potencial de sua criação, sua patente foi adquirida pela empresa americana Westinghouse, contudo, seu projeto foi descontinuado em via do seu baixo rendimento (ALEX MARTINS DUARTE et al., 2024).

Foi em 1889, com Michael Von Dolivo-Dobrovolski que a criação do motor de indução com uma potência de 80W, um rendimento médio de 80% com ótimo conjugado de partida, apresentado a empresa alemã AEG. Seu projeto foi continuado, com o desenvolvimento de motores com potências de 0,4 a 7,5 kW (ALEX MARTINS DUARTE et al., 2024).

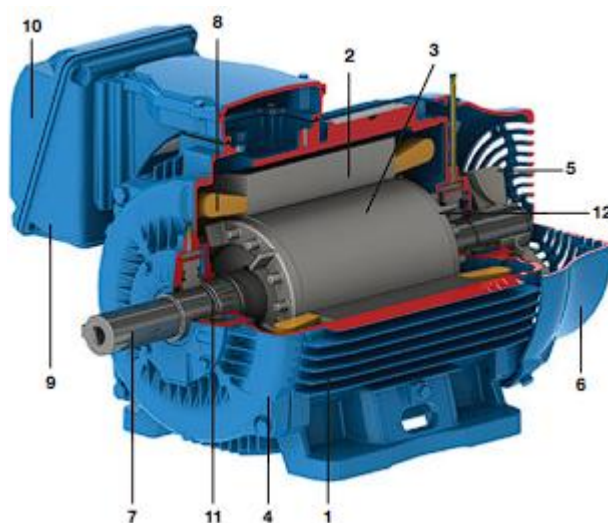
Motores elétricos de corrente alternada são mais usados pela disponibilidade da distribuição de energia elétrica ocorrer em corrente alternada. Em relação ao motor de indução, o seu uso é ainda mais comum que o motor síncrono, com 90% do consumo das indústrias em função desses motores (ALEXANDRE ABRAHAMI PINTO DA CUNHA, 2023). Seu preço para a mesma potência é menor, sua robustez mecânica é maior e sua construção é simples (WEG, 2024).

Figura 20 - Diagrama de motores elétricos.



Fonte: (WEG, 2024)

Figura 21 - Visão em corte de um motor elétrico de indução.



Fonte: (WEG, 2024)

Um motor de indução trifásico é fundamentalmente dividido em duas partes principais: estator e rotor. O estator consiste na parte estática do motor, a sua carcaça, responsável pela integridade física e pela dissipação de calor. O rotor é a parte móvel do motor, girando através da indução gerada a partir do campo girante do estator.

Na figura 21, são demonstradas detalhadamente os aspectos construtivos do motor elétrico.

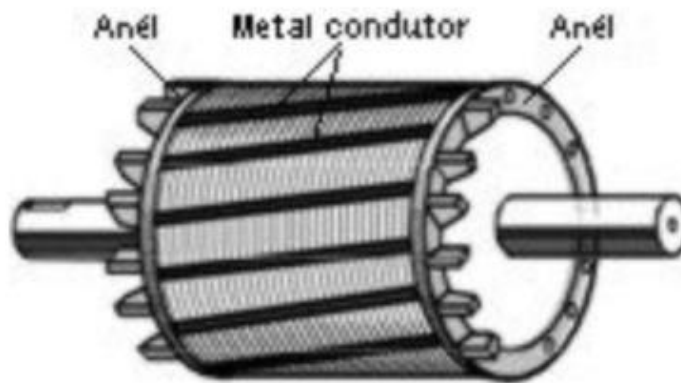
- Carcaça (1)
- Núcleo de chapas do estator (2)
- Núcleo de chapas do rotor (3)
- Tampa (4)
- Ventilador (5)
- Tampa defletora (6)
- Eixo (7)
- Enrolamento trifásico (8)
- Caixa de ligação (9)
- Terminais (10)
- Rolamentos (mancais) (11)

Há dois principais tipos de rotores: gaiola de esquilo e rotor bobinado. O rotor em forma de gaiola de esquilo consiste em barras metálicas paralelas alojados nas ranhuras das chapas

laminadas ligadas, em suas extremidades, por anéis condutores em curto-circuito. A simplicidade de sua construção junto de sua robustez, contribui para essa ser a construção mais usual dos motores de indução (STEPHAN D. UMANS, 2014).

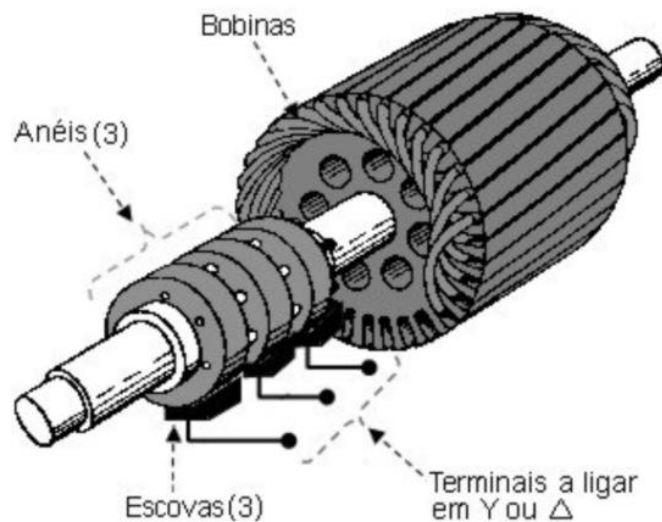
O outro tipo, o rotor bobinado, tem a sua construção similar ao estator do motor, com os seus terminais conectados a anéis deslizantes montados sob o eixo. Escovas de carvão são apoiadas sobre esses anéis, permitindo que os terminais do rotor se tornem disponíveis externamente ao motor (STEPHAN D. UMANS, 2014).

Figura 22 - Rotor gaiola de esquilo.



Fonte: (ANTONIO M. S. FRANCISCO, 2006)

Figura 23 - Rotor bobinado.

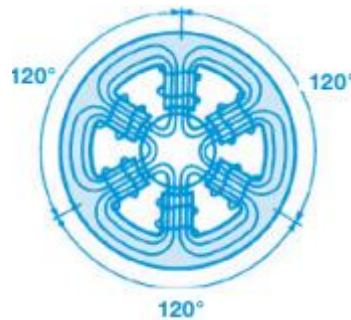


Fonte: (ANTONIO M. S. FRANCISCO, 2006)

### 2.5.1 Princípio de funcionamento

O princípio de funcionamento básico de um motor elétrico é baseado na interação do campo magnético gerado pela corrente presente nos enrolamentos do ESTATOR com o campo magnético gerado pela corrente presente nos enrolamentos do ROTOR (ALEXANDRE ABRAHAMI PINTO DA CUNHA, 2023). O motor elétrico de indução é alimentado em corrente alternada (CA). Com isso, dada a configuração espacial dos enrolamentos distribuídos no motor, o campo magnético produzido pelo estator é girante (ANTONIO M. S. FRANCISCO, 2006).

Figura 24 - Enrolamentos trifásicos do estator.



Fonte: (WEG, 2024)

O campo magnético girante do estator ( $B_s$ ), em contato com o rotor, induz uma tensão nos enrolamentos do rotor ( $E_r$ ), em conformidade com a Lei de Faraday, conforme a equação (23). Essa tensão induzida gera um fluxo de corrente elétrica no rotor ( $I_r$ ), que flui em sentido contrário à causa que lhe deu origem, em respeito à Lei de Lenz. Essa corrente produz um campo magnético girante ( $B_r$ ). A interação que ocorre entre o campo magnético do estator e do rotor faz com que ocorra o movimento (ANTONIO M. S. FRANCISCO, 2006; STEPHEN J. CHAPMAN, 2013).

$$e_{ind} = -N \frac{d\phi}{dt} \quad (22)$$

No rotor, o campo magnético gerado tende a se opor ao campo magnético do estator, gerando um polo contrário. Como polos contrários se atraem, o campo magnético do rotor, sempre tenta acompanhar o campo magnético do estator. Assim como diz a Lei de Laplace, um condutor, percorrido por corrente, mergulhado num campo magnético, fica sujeito a uma força eletromagnética (ANTONIO M. S. FRANCISCO, 2006).

---


$$F = B.I.sen\alpha \quad (23)$$

A velocidade de rotação do campo magnético girante de um motor é chamada de velocidade síncrona ( $ns$ ), expressada, em rpm (rotações por minuto), por:

$$ns = \frac{120.fs}{P} \quad (24)$$

Sendo  $fs$  a frequência da rede aplicada ao estator, em Hertz (Hz), e  $P$  o número de polos. Sabe-se que a corrente gerada no rotor é induzida, e depende da diferença de velocidade do rotor em relação ao campo girante, uma vez que a velocidade do rotor é ligeiramente inferior a velocidade do campo girante. Assim, um termo comum para definir essa diferença de velocidade é chamado de escorregamento, dado pela seguinte equação:

$$s = \frac{ns-nm}{ns} \cdot 100\% \quad (25)$$

Sendo,  $ns$  a velocidade dos campos magnéticos e  $nm$  a velocidade mecânica do eixo do motor.

Para o caso de um motor com o seu eixo estático ( $nm = 0$ ), o seu escorregamento é unitário ( $s = 1$ ). E para o caso de a velocidade do eixo girar na velocidade síncrona do campo magnético ( $nm = ns$ ), o seu escorregamento é zero ( $s = 0$ ), o que não deve ocorrer, pois assim, a corrente induzida no rotor seria zero ( $Ir = 0$ ), e conseqüentemente, o conjugado seria nulo. Para expressar adequadamente a velocidade mecânica, pode-se fazer:

$$nm = (1 - s).ns \quad (26)$$

Em velocidades intermediárias as supracitadas, a frequência do rotor é diretamente proporcional à diferença entre a velocidade do campo girante e a velocidade do rotor. Essa frequência pode ser expressa como sendo:

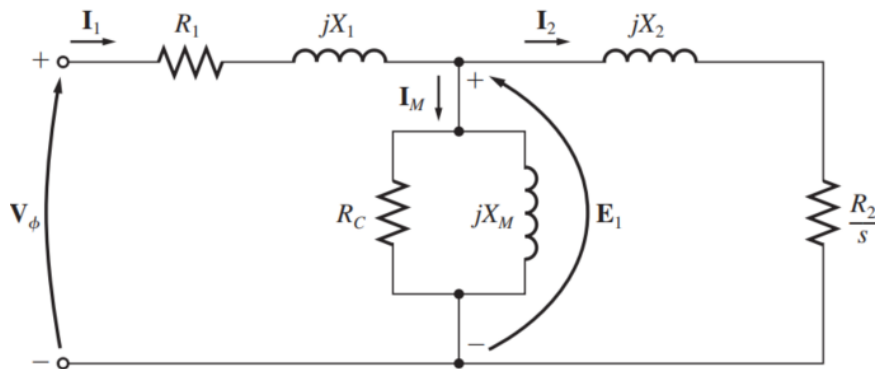
$$fr = sfs \quad (27)$$

Essa frequência  $fr$  pode ser denominada como frequência de escorregamento do rotor. Sendo o movimento relativo entre o fluxo do estator e os condutores do rotor responsáveis por induzir tensões de frequências  $fr$ , uma máquina de indução é por vezes referida como um transformador de frequência variável (STEPHAN D. UMANS, 2014; STEPHEN J. CHAPMAN, 2013)

## 2.5.2 Característica torque-velocidade

A característica torque-velocidade de um motor é muito importante do ponto de vista das suas aplicações com controle de velocidade. Para isso, deve ocorrer a análise do circuito equivalente do motor de indução.

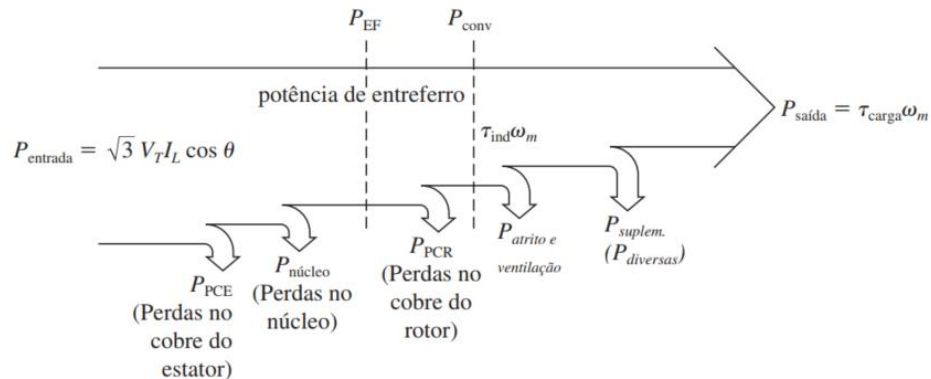
Figura 25 - Circuito equivalente por fase de um motor de indução.



Fonte: (STEPHEN J. CHAPMAN, 2013)

Na alimentação do estator do motor de indução, a potência de entrada acontece na forma das tensões e correntes trifásicas da rede elétrica. Na saída do motor de indução, no eixo do rotor, a potência ocorre de forma mecânica. Entre essas potências de entrada e saída, há presente as perdas do motor. Essas perdas podem ser visualizadas na figura 26 abaixo.

Figura 26 - Diagrama do fluxo de potência de um motor de indução.

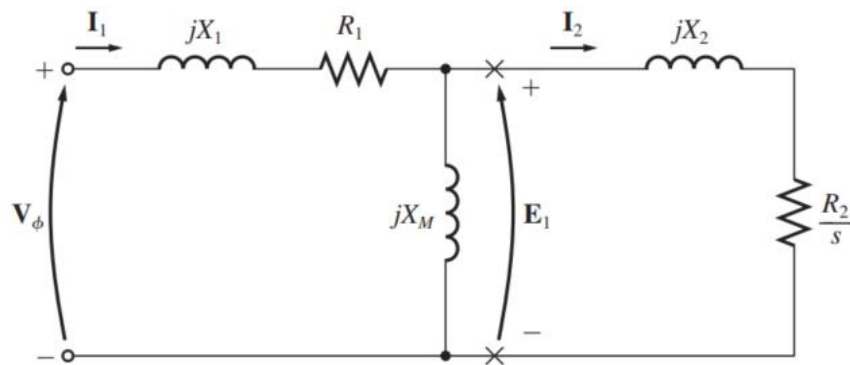


---

Fonte: (STEPHEN J. CHAPMAN, 2013)

Uma forma simples de se calcular o circuito equivalente da figura 29, é utilizando o Teorema de Thevenin. Sabe-se que, de acordo com este teorema, qualquer circuito linear pode ser substituído por uma única fonte de tensão em série com uma impedância equivalente. Fazendo isso com o circuito da figura 27, tem-se:

Figura 27 - Circuito equivalente do motor de indução - Teorema de Thevenin.



Fonte: (STEPHEN J. CHAPMAN, 2013)

O circuito da figura 27, representa a simplificação do circuito equivalente base do motor de indução, com  $R_m$  desprezado devido ao seu valor alto comparado com  $X_m$ . Assim, pode-se encontrar o equivalente de Thevenin à esquerda dos pontos marcados, ou seja, obter o equivalente de Thevenin do circuito do Estator, como sendo:

$$R_{th} = R_1 \cdot \left( \frac{X_m}{X_1 + X_m} \right)^2 \quad (28)$$

$$X_{th} \cong X_1 \quad (29)$$

A partir do circuito equivalente na figura 27, pode-se determinar o módulo da corrente do rotor  $I_2$ , como sendo:

$$I_2 = \frac{V_s}{\sqrt{\left( R_{th} + \frac{R_2}{s} \right)^2 + (X_{th} + X_2)^2}} \quad (30)$$

Sabe-se que a potência no entreferro, conhecida como potência eletromagnética, sendo essa a potência que cruza a lacuna de ar entre o estator e o rotor, pode ser dada como:

$$P_{ef} = 3 \cdot I_2^2 \cdot \frac{R_2}{s} \quad (31)$$

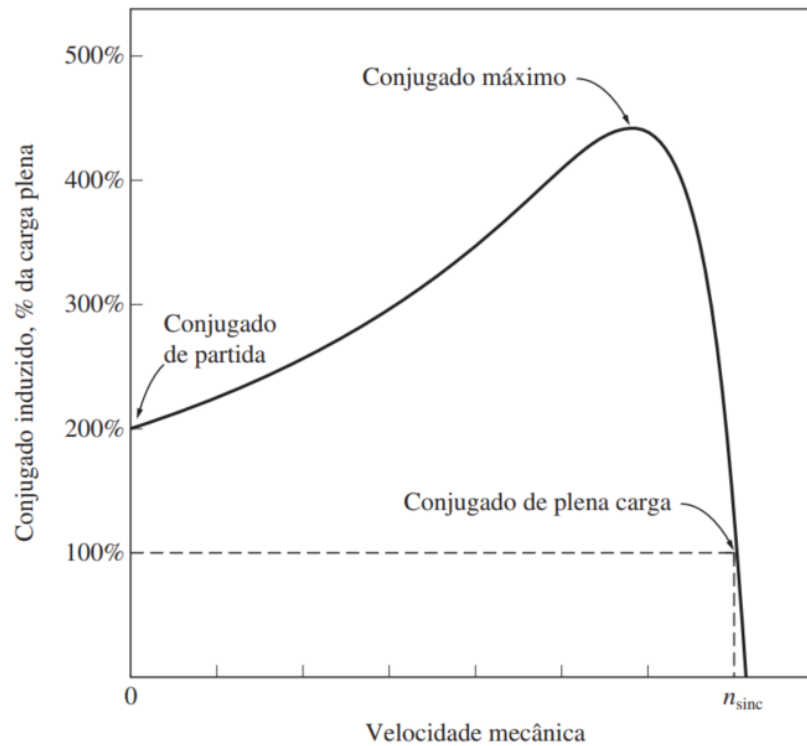
Substituindo a equação (I2) em (Pef), e sabendo que o Torque Induzido é dado por:

$$T_{ind} = \frac{P_{ef}}{\omega_{sinc}} \quad (32)$$

A nova equação do conjugado induzido, relacionando-a com o circuito equivalente do motor, é:

$$T_{ind} = \frac{3.V_{th}^2.R_2}{s.\omega_{sinc}.\left(\left(R_{th}+\frac{R_2}{s}\right)^2+(X_{th}+X_2)^2\right)} \quad (33)$$

Figura 28 - Curva característica de conjugado x velocidade.



Fonte: (STEPHEN J. CHAPMAN, 2013)

Observe pela figura 28 que, na velocidade síncrona, o conjugado de um motor de indução é zero.

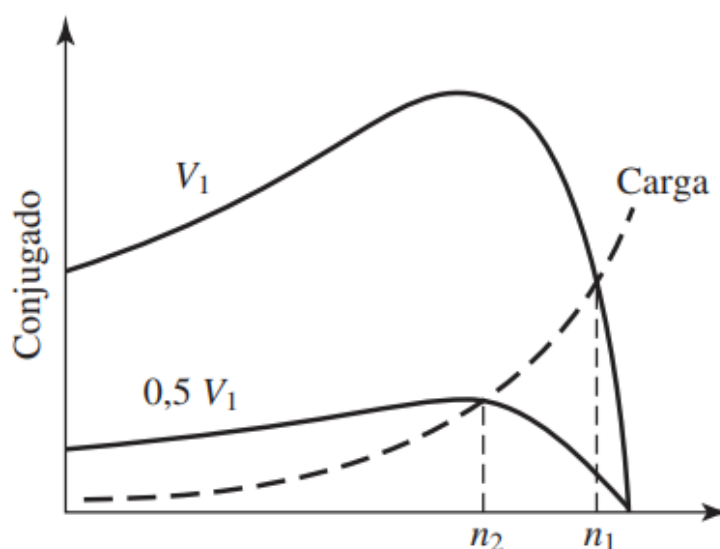
### 2.5.3 Controle de velocidade

Antes do advento de equipamentos modernos como os inversores, os motores de indução, no geral, não eram boas máquinas para se controlar a velocidade. A sua faixa de variação era

estreita, limitada a menos 5% do escorregamento e proporcional a carga acoplada no eixo do motor (STEPHEN J. CHAPMAN, 2013) As técnicas para variação da velocidade de um motor de indução consistem em: mudar o número de polos da máquina, variar a frequência de linha, variar a tensão de linha, variar a resistência do rotor e aplicar tensões de determinadas frequências ao rotor (STEPHAN D. UMANS, 2014).

Mudar a tensão de alimentação do estator do motor de indução, implica em alterar o seu torque induzido, que é proporcional ao quadrado da tensão. Este é um método de controle comum em pequenos motores com rotores de gaiola de esquilo, para acionar pequenas cargas de bombas e ventiladores. Sua principal desvantagem fica em questão do seu baixo rendimento, especialmente em baixas velocidades (DANIEL W. HART, 2011; STEPHAN D. UMANS, 2014)

Figura 29 - Controle de velocidade pela tensão de linha.



Fonte: (STEPHAN D. UMANS, 2014)

Variar a frequência de linha de um motor de indução significa mudar proporcionalmente a sua velocidade síncrona, fato este confirmado pela equação (24). O controle pode ser feito tanto para reduzir ou aumentar a velocidade da máquina, no entanto, existem algumas implicações.

Para valores de frequência reduzidos, a velocidade síncrona e a reatância equivalente do motor se reduzem, ocasionando um aumento do seu conjugado, assim como do fluxo no núcleo magnético. Como a magnitude da tensão terminal permanece constante, problemas de saturação do núcleo podem ocorrer, devido a presença de corrente elevadas. Para valores de frequência

---

elevados, a velocidade síncrona e a reatância equivalente aumentam, o que resulta em uma redução do conjugado, assim como do fluxo no núcleo magnético.

Isso pode ser observado tanto na equação de conjugado induzido ( $T_{ind}$ ), quanto na equação do fluxo (34), resultante da equação de tensão induzida (22):

$$\phi(t) = -\frac{V_m}{\omega \cdot N_p} \cdot \cos \omega t \quad (34)$$

A proporcionalidade das equações do torque induzido, dado pela equação (32) com o fluxo resultante, dado pela equação (34), é confirmado pela relação:

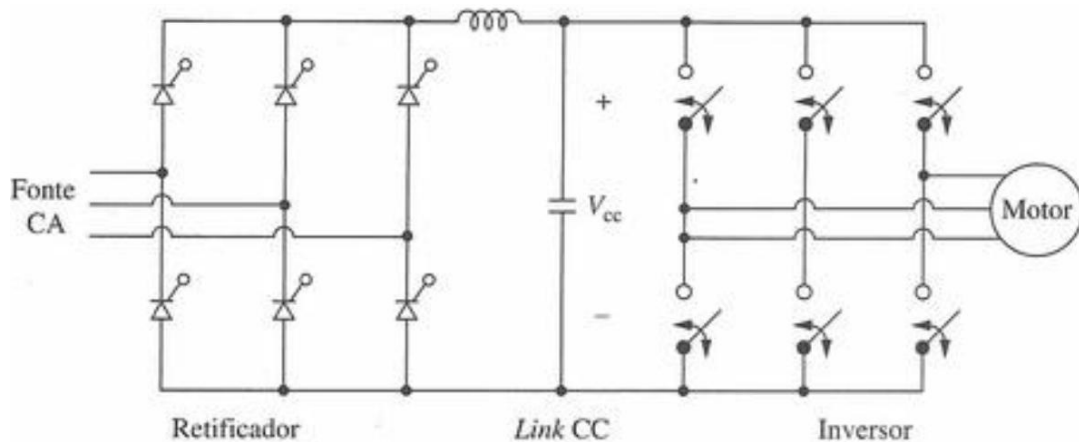
$$T_{ind} = K \cdot \phi \cdot I_2 \quad (35)$$

#### 2.5.4 Inversor de Frequência

Um método amplamente usado nas indústrias para acionamento e controle de velocidade dos motores de indução, atualmente, consiste no uso dos inversores de frequência, também conhecidos como conversores estáticos de frequência, ou inversores de tensão (WEG, 2024). Seu uso é fundamental em aplicações com velocidades variadas, uma vez que ele permite a operação do motor de modo eficiente, funcionando de forma adequada em cada momento solicitado. (ALEXANDRE ABRAHAMI PINTO DA CUNHA, 2023).

Os inversores ainda disponibilizam uma gama de informações importantes sobre o motor acionado, fornecendo dados de corrente, tensão, frequência, velocidade, conjugado e temperatura. Essas informações credibilizam o uso desse dispositivo elétrico, pois garante maior vida útil ao motor (ALEXANDRE ABRAHAMI PINTO DA CUNHA, 2023). Dentre os benefícios de seu uso, estão a redução de custos, o controle à distância, a versatilidade, o aumento da qualidade e da produtividade e o uso eficiente da energia (WEG, 2024).

Figura 30 - Diagrama básico de um inversor de frequência.

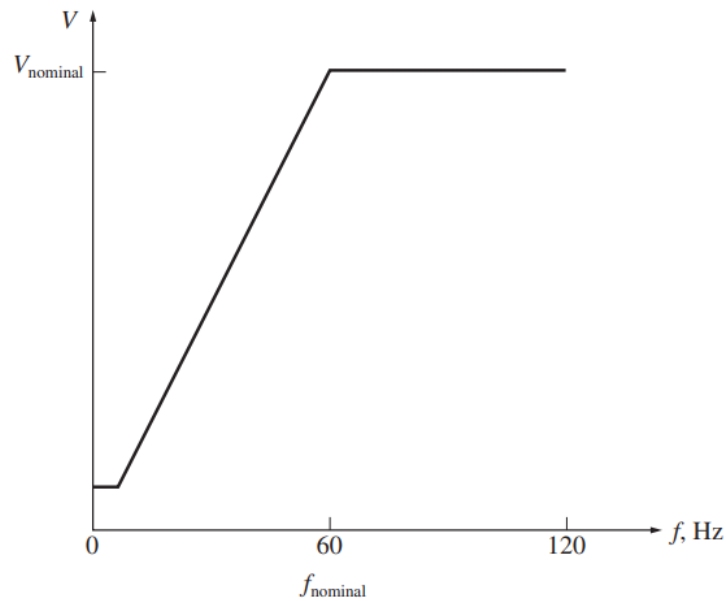


Fonte: (DANIEL W. HART, 2011)

Uma forma fácil de controlar a velocidade de um motor de indução é variando a sua frequência de alimentação. Para isso, os inversores produzem uma tensão CC ajustável, através do retificador via ponte de diodos, e em seguida, regulam a tensão retificada com um filtro de capacitores e então, essa tensão CC é convertida em uma tensão CA na frequência desejada, através de chaves eletrônicas, como os transistores IGBT, por meio de técnica de largura de pulso (PWM). (DANIEL W. HART, 2011; WEG, 2024).

Para um controle adequado de velocidade utilizando o inversor de frequência, um outro método de controle deve ser inserido ao controle pela frequência: o controle de tensão. Este método deve variar linearmente com a frequência, de modo que o conjugado, e a densidade de fluxo no entreferro sejam aproximadamente constantes. Essa técnica de controle é conhecida como controle escalar (STEPHAN D. UMANS, 2014; WEG, 2024).

Figura 31 - Curva característica da tensão em função da frequência de alimentação – Padrão de uso geral.



Fonte: (STEPHEN J. CHAPMAN, 2013)

Observando a figura 31, a relação  $V/f$  (volts por hertz) é constante até a frequência nominal (ou frequência base) do motor de indução, ou seja, a tensão cresce proporcionalmente com a frequência. A partir da frequência nominal, o valor de tensão deve ser mantido constante, para respeitar os limites de isolamento dos enrolamentos do estator.

---

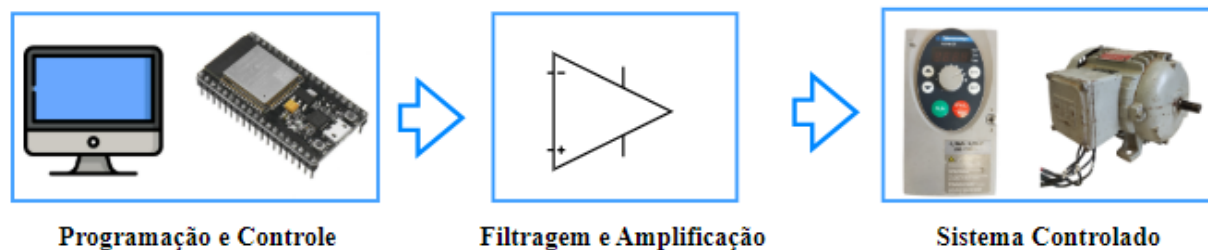
### 3 SISTEMA PROPOSTO

---

Neste capítulo, serão apresentados os materiais utilizados na confecção do projeto de controle de velocidade do motor de indução, abrangendo tanto os materiais físicos quanto os softwares de programação empregados.

O desenvolvimento e a aplicação do projeto foram realizados no Laboratório de Inteligência Artificial, Eletrônica de Potência e Sistemas Digitais da UFMS, o BATLAB. A maior parte dos materiais utilizados foi fornecida por esse laboratório.

Figura 32 – Diagrama geral do sistema proposto.



Fonte: Próprio autor.

#### 3.1 ARDUINO IDE E VISUAL STUDIO CODE

O software de programação utilizado para enviar e gravar o código no ESP32 foi o Arduino IDE, um software de código aberto que auxilia desenvolvedores na escrita e no upload de códigos em placas Arduino.

Sua interface é responsiva e moderna, contando com recursos de gerenciamento de placas e bibliotecas, monitor serial, ferramentas de edição e compilação de código, além de suporte às linguagens C e C++. Apesar de ser pensado para placas Arduino, o microcontrolador ESP32 pode ser programado no Arduino IDE mediante a utilização de uma biblioteca fornecida pela própria Espressif.

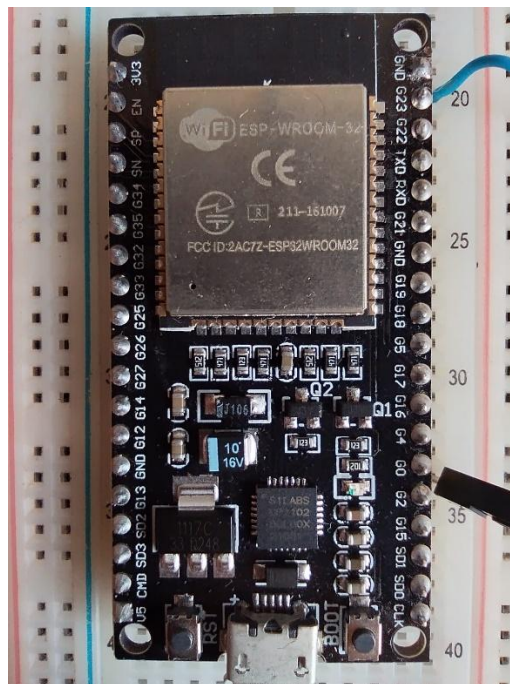
A escrita do código HTML5, em conjunto com o CSS3 e a linguagem de programação JavaScript para criação do servidor web do controlador de velocidade, foi realizada no editor

Visual Studio Code (VSC). Posteriormente, o código desenvolvido no VSC foi incorporado ao ambiente do Arduino IDE.

### 3.2 MICROCONTROLADOR ESP32 DO BATLAB

O microcontrolador escolhido para o controle da velocidade do motor foi o ESP32, em razão de seu baixo custo e facilidade de integração com o inversor utilizado, além da capacidade de comunicação sem fio por Wi-Fi ou Bluetooth. A programação do ESP32 foi realizada por meio do Arduino IDE.

Figura 33 - ESP32 do BATLAB.



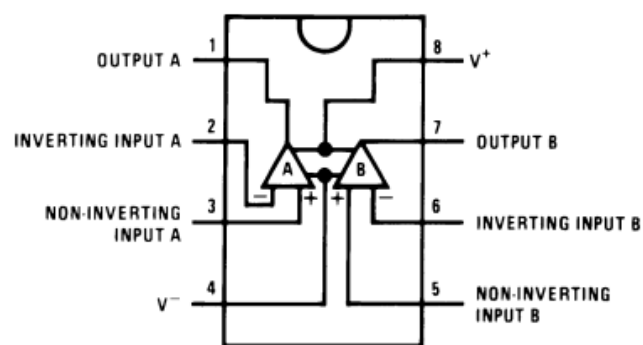
Fonte: Próprio autor.

O ESP32 possui 16 canais independentes que podem ser configurados para gerar sinais PWM. Esses sinais podem ser acessados por qualquer porta GPIO, com exceção das GPIO34 e GPIO39. A resolução do sinal PWM é de 8 bits, com valores variando de 0 (0 V) a 255 (3,3 V) (BLOG SMART KITS, 2020; ESPRESSIF SYSTEMS, 2024). A comunicação com o computador para upload do código é feita por meio de um cabo USB.

### 3.3 CIRCUITO DE FILTRAGEM E AMPLIFICAÇÃO

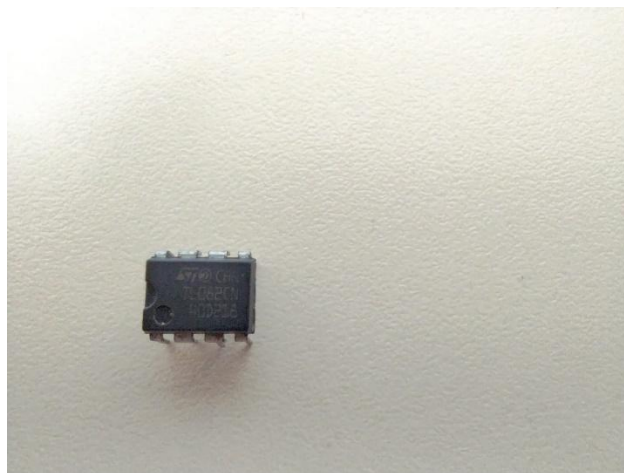
Para filtrar e amplificar o sinal de tensão de saída do ESP32, foi utilizado o Circuito Integrado (CI) TL082, que contém dois amplificadores operacionais (AmpOp) internos, conforme ilustrado na figura 34.

Figura 34 - Circuito interno



Fonte: (TEXAS INSTRUMENTS, 1998).

Figura 35 - AmpOp TL082 do BATLAB.



Fonte: Próprio autor.

Tabela 5 - Características principais do AmpOp tl082.

| Parâmetro             | Valor     |
|-----------------------|-----------|
| Tensão de alimentação | $\pm 18V$ |

|                         |                 |
|-------------------------|-----------------|
| Ganho de tensão         | 100             |
| Tensão de entrada       | 15V             |
| Faixa de frequência     | 4MHz            |
| Impedância de entrada   | $10^{12}\Omega$ |
| Temperatura de operação | 0 a 70°C        |
| Tj(max)                 | 150°C           |

Fonte: (TEXAS INSTRUMENTS, 1998).

### 3.4 INVERSOR DE FREQUÊNCIA DO BATLAB

O inversor de frequência utilizado para o controle de velocidade do motor de indução foi o Telemecanique Altivar 31, modelo ATV31H018M2A, indicado para acionamento de máquinas com potência aproximada de 0,25 HP (ou 0,18 kW) e faixa de tensão de 200 a 240 V. A figura 36 apresenta o inversor.

Figura 36 - Inversor de frequência do BATLAB.



Fonte: Próprio autor.

A tensão de alimentação do inversor é monofásica, com valores de 200 a 240 V a 50/60 Hz. Os dois bornes de alimentação (R/L1 e S/L2) e o borne de aterramento são conectados com cabos de 2,5 mm<sup>2</sup> de espessura. A corrente de linha é de 3 A para tensão de 200 V e 2,5 A para tensão de 240 V. A saída para o motor de indução é feita por três bornes (U/T1, V/T2 e W/T3), configurados em ligação delta ao inversor de frequência.

Tabela 6 - Descrições dos bornes de potência do inversor do BATLAB.

| Borne | Função                           |
|-------|----------------------------------|
| Terra | Borne de aterramento             |
| R/L1  | Alimentação de potência          |
| S/L2  |                                  |
| U/T1  | Saída para o motor               |
| V/T2  |                                  |
| W/T3  |                                  |
| P0    | Polaridade + barramento CC       |
| PA/+  | Saída - resistência de freio (+) |
| PB    | Saída - resistência de freio     |
| PC/-  | Polaridade - barramento CC       |

Fonte: (SCHNEIDER ELECTRIC, [s.d.]).

Figura 37 - Bornes de potência do inversor do BATLAB.



Fonte: Próprio autor.

O controle de velocidade do motor é realizado por meio dos bornes de controle do inversor, sendo a comunicação com o microcontrolador ESP32 feita em uma entrada analógica de tensão.

Tabela 7 - Descrições dos bornes de potência do inversor do BATLAB.

| Borne | Função                               | Característica |
|-------|--------------------------------------|----------------|
| COM   | Comum das entradas/saídas analógicas | 0V             |
| AI1   | Entrada analógica em tensão          | 0 - 10V        |
| 10V   | Alimentação para potenciômetro       | 10V            |
| AI2   | Entrada analógica em tensão          | $0 \pm 10V$    |
| AI3   | Entrada analógica em corrente        | 0 - 20mA       |

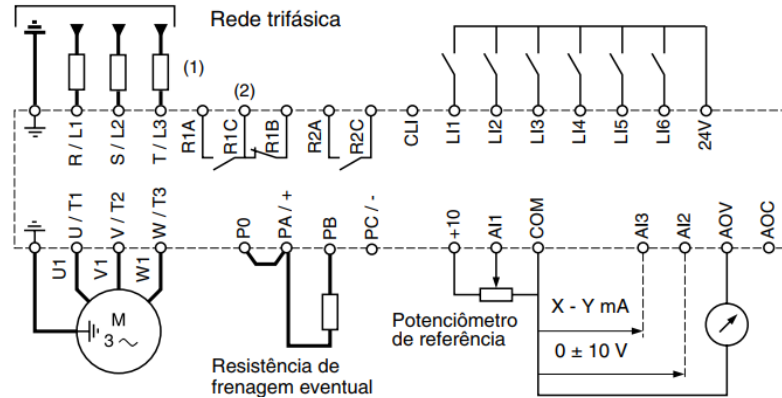
Fonte: (SCHNEIDER ELECTRIC, [s.d.]).

Figura 38 - Bornes de controle do inversor do BATLAB.



Fonte: Próprio autor.

Figura 39 - Diagrama do inversor.



Fonte:(SCHNEIDER ELECTRIC, [s.d.]).

### 3.5 MOTOR DE INDUÇÃO DO BATLAB

O motor de indução trifásico utilizado já estava presente no laboratório BATLAB antes da elaboração do projeto. Ele estava conectado ao pequeno inversor de frequência mostrado anteriormente. A figura 44 apresenta o motor.

Figura 40 - Motor de indução do BATLAB.



Fonte: Próprio autor.

Este motor foi fabricado pela empresa WEG, fundada em 1961, em Jaraguá do Sul - SC, uma das maiores fabricantes de equipamentos elétricos do mundo. A figura 41 exibe informações

básicas do motor, como frequência, potência, velocidade mecânica, tensões e correntes para as configurações delta e estrela, entre outros dados.

Figura 41 - Especificações técnicas do motor de indução.



Fonte: Próprio autor.

---

## 4 RESULTADOS

---

Neste capítulo, são explicados o desenvolvimento e o funcionamento do sistema, bem como apresentados os resultados práticos obtidos em laboratório.

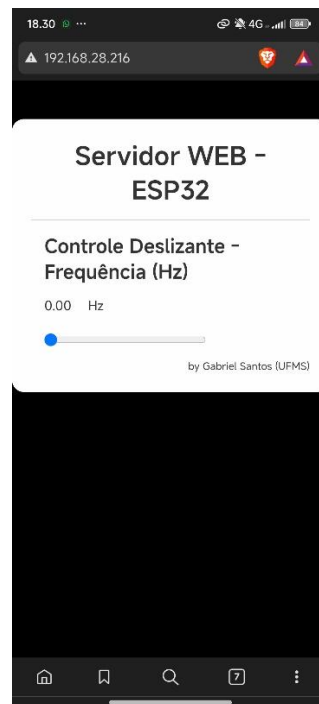
Para fins de organização e entendimento do projeto, este capítulo foi dividido em etapas, seguindo a ordem cronológica de construção.

### 4.1 PRIMEIRA ETAPA

A primeira etapa do projeto consistiu no desenvolvimento da programação no software Arduino IDE, com a parte do código referente ao HTML desenvolvida no Visual Studio Code (VSC). Esse código foi enviado ao microcontrolador ESP32 para realizar o controle proporcional do ciclo de trabalho do sinal PWM gerado em uma das GPIOs do ESP32. Esse sinal varia de 0 a 255, o que altera a tensão de saída do ESP32 entre 0 e 3,3 V.

Assim que o código é enviado e armazenado no ESP32, no monitor serial presente na interface do Arduino IDE, é mostrado o endereço IP gerado pelo ESP32. Esse endereço IP, deve ser copiado e anexado em um navegador web, de forma que se possa acessar o servidor web com o controle deslizante. Qualquer aparelho, desde que esteja conectado na mesma rede, poderá acessar o servidor web através do endereço IP.

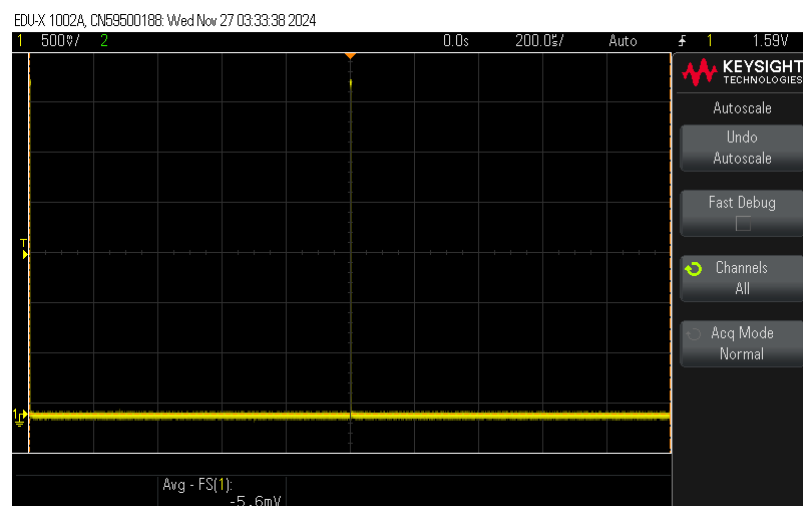
Figura 42 - Interface do servidor web.



Fonte: Próprio autor.

Na figura 46, observa-se há um controle deslizante presente na interface do servidor, onde é possível mudar a frequência de alimentação. Para isso, o código implementa um controle proporcional do sinal PWM. Assim, enquanto o sinal PWM varia de 0 a 255, o sinal de frequência é ajustado de 0 a 60 Hz, de forma proporcional, em conformidade com o registrado nos inversores de frequência.

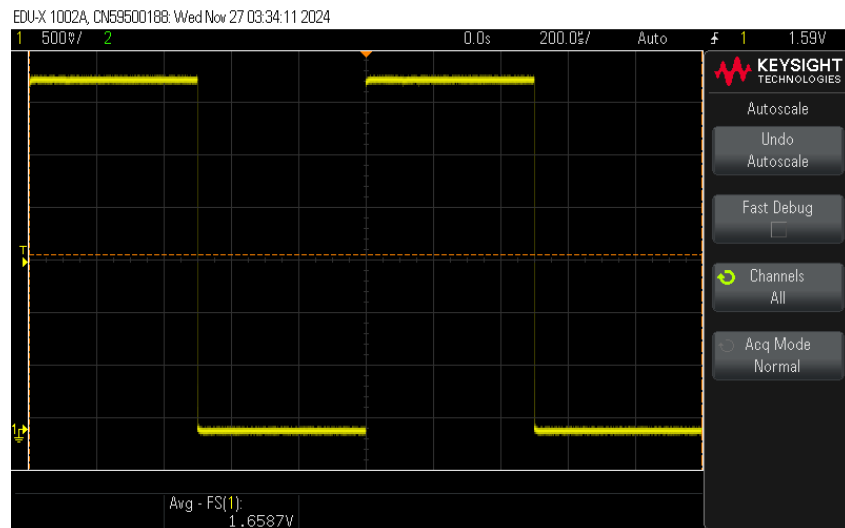
Figura 43 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 0Hz



---

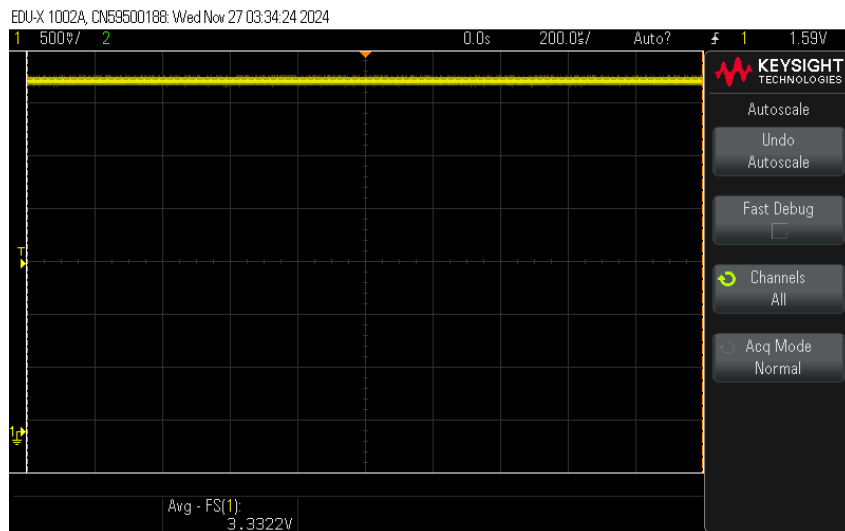
Fonte: Próprio autor.

Figura 44 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 30Hz



Fonte: Próprio autor.

Figura 45 - Sinal de tensão de saída da porta GPIO2 do ESP32 – 60Hz



Fonte: Próprio autor.

A programação detalhada escrita no ambiente do Arduino IDE está anexada no apêndice, ao final deste documento.

---

## 4.2 SEGUNDA ETAPA

A segunda etapa do sistema consistiu no projeto de um filtro passa-baixa (PB) de segunda ordem, conectado à saída de tensão em uma das portas GPIO do ESP32.

A topologia do circuito do filtro é do tipo Sallen-Key, e o formato da resposta em frequência escolhido foi o de Butterworth, caracterizado por uma resposta de magnitude plana nas faixas de passagem e de rejeição, com uma região de transição moderada.

Os dados iniciais escolhidos para este filtro são:

- Frequência de corte ( $f_c$ ) de 50Hz;
- Ganho unitário;
- Coeficiente real  $a_1$ : 1,4142;
- Coeficiente real  $b_1$ : 1,0000;
- Capacitor  $C_1$ :  $39\eta F$

O capacitor  $C_2$  deve ser escolhido com base na equação (18), que define o valor mínimo necessário para o circuito.

$$C_{2\min} = C_1 \cdot \frac{4 \cdot b_1}{a_1^2} = 39\eta \cdot \frac{4 \cdot 1}{1,4142^2} = 78\eta F$$

O capacitor  $C_2$  escolhido para o filtro PB de Butterworth foi de  $100\eta F$ .

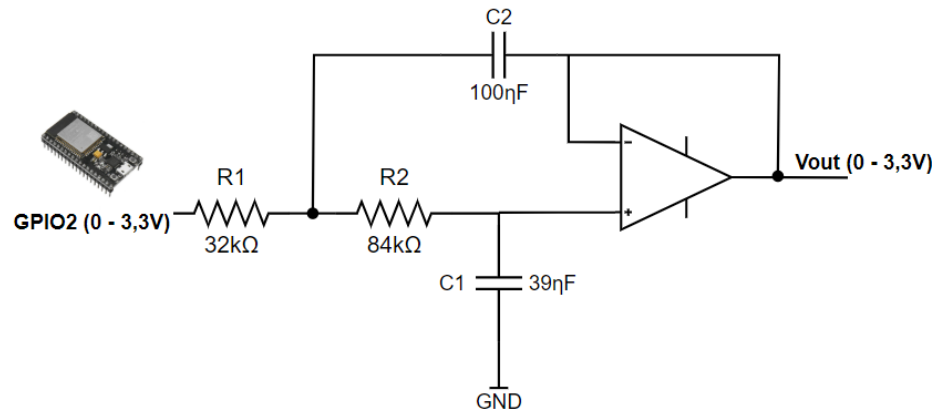
Com os valores dos capacitores  $C_1$  e  $C_2$  definidos, os resistores  $R_1$  e  $R_2$  foram calculados por meio da equação (17).

$$R_1 = \frac{1,4142 \cdot 100\eta - \sqrt{(78\eta)^2 \cdot (100\eta)^2 - 4 \cdot 1 \cdot 39\eta \cdot 100\eta}}{4 \cdot \pi \cdot 50 \cdot 78\eta \cdot 100\eta} = 30,644k\Omega$$

$$R_2 = \frac{1,4142 \cdot 100\eta + \sqrt{(78\eta)^2 \cdot (100\eta)^2 - 4 \cdot 1 \cdot 39\eta \cdot 100\eta}}{4 \cdot \pi \cdot 50 \cdot 78\eta \cdot 100\eta} = 84,78k\Omega$$

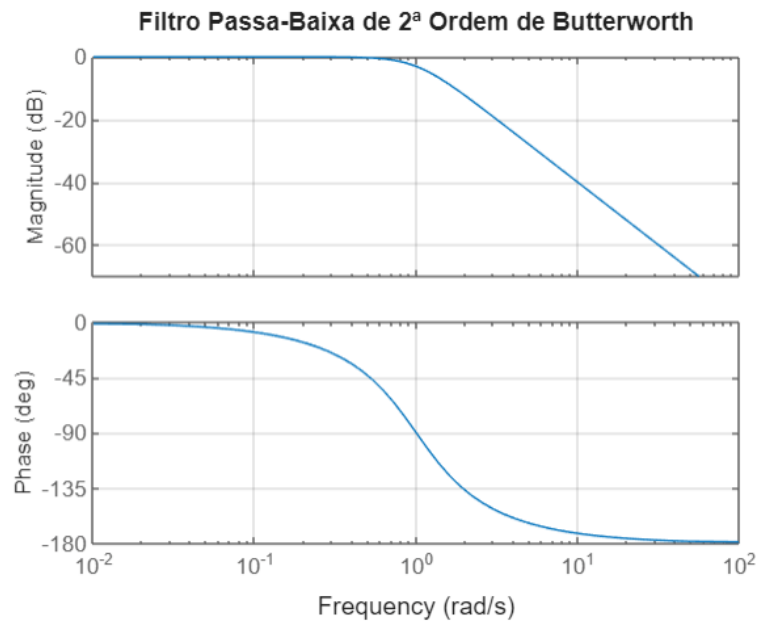
Os valores comerciais adotados para a montagem do filtro foram de  $32k\Omega$  e  $84k\Omega$ , respectivamente.

Figura 46 - Diagrama elétrico do filtro PB de 2ª ordem de Butterworth com topologia Sallen-Key.



Fonte: Próprio autor.

Figura 47 - Diagrama de Bode da resposta em frequência do filtro.



Fonte: Próprio autor.

O código com os cálculos dos componentes do filtro PB de 2ª ordem de Butterworth está disponível no apêndice, ao final desse documento.

### 4.3 TERCEIRA ETAPA

Na terceira etapa, após o sinal ser filtrado, foi necessário implementar um circuito de amplificação. Esse circuito foi utilizado devido à diferença de tensão entre o ESP32, que varia de 0 a 3,3 V, e o circuito de controle do inversor de frequência, que opera com tensões de 0 a 10 V para controlar a velocidade do motor.

Para resolver essa incompatibilidade, utilizou-se um circuito amplificador não inversor com realimentação negativa, que não altera a fase do sinal de tensão de saída (ANTONIO PERTENCE JR., 2014; WENDLING, 2010).

A expressão matemática correspondente a este circuito é:

$$V_o = \left(1 + \frac{R_f}{R_3}\right) \cdot V_i \quad (36)$$

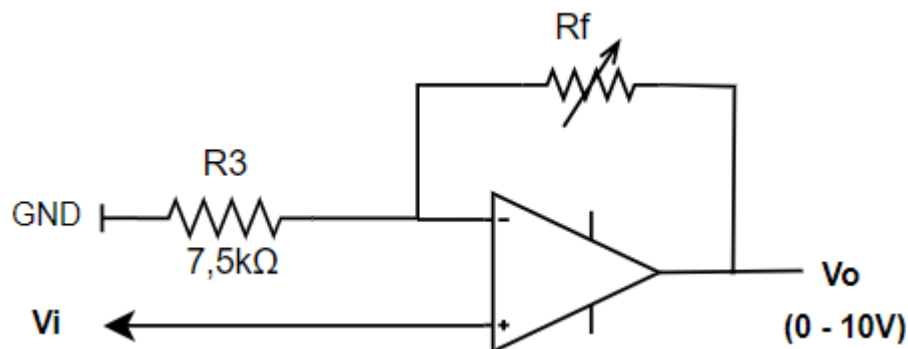
Para alcançar o ganho necessário, a parcela correspondente ao ganho linear do AmpOp foi ajustada para aproximadamente 3. Assim, o sinal de saída amplificado foi:

$$V_o = \left(1 + \frac{15k}{7,5k}\right) \cdot 3,3 \cong 10V$$

O ganho em decibéis foi calculado como:

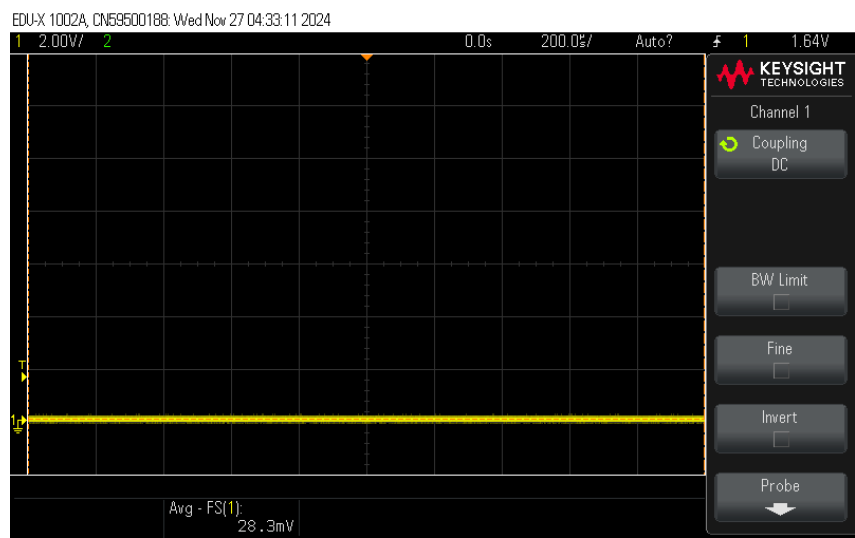
$$G_{dB} = 20 \cdot \log(3) = 9,542dB$$

Figura 48 - Diagrama elétrico do circuito amplificador não inversor.



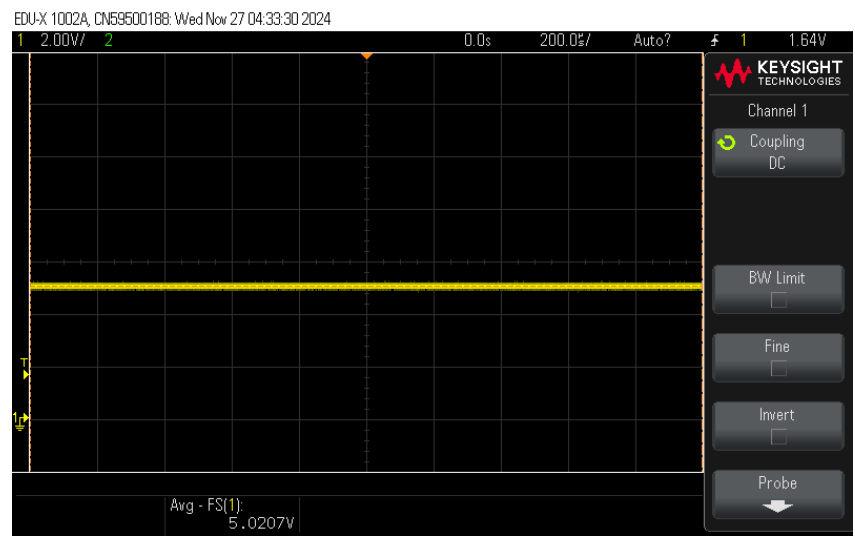
Fonte: Próprio autor.

Figura 49 - Sinal de tensão de saída filtrado e amplificado – 0Hz



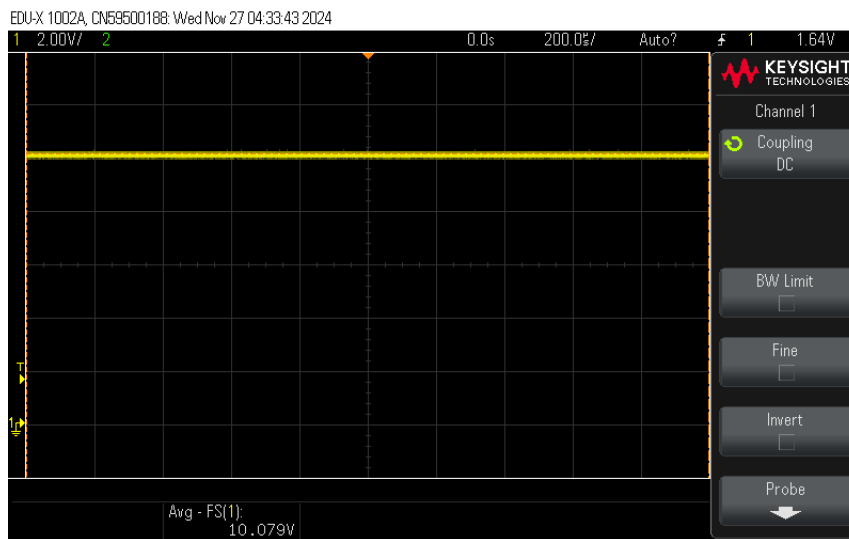
Fonte: Próprio autor.

Figura 50 - Sinal de tensão de saída filtrado e amplificado – 30Hz



Fonte: Próprio autor.

Figura 51 - Sinal de tensão de saída filtrado e amplificado – 60Hz



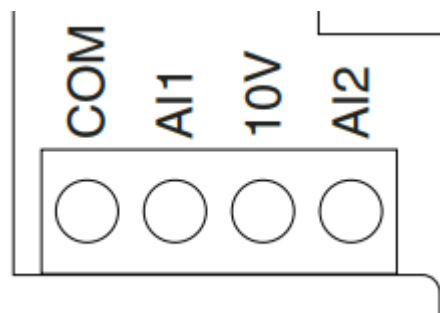
Fonte: Próprio autor.

#### 4.4 QUARTA ETAPA

Na quarta etapa, o sinal de tensão amplificado foi conectado ao borne de entrada analógica de tensão AI2 do inversor, além do borne comum de terra (COM). O sinal de entrada analógica AI2 pode variar de 0 a 10 V, resultando na alteração direta da frequência de alimentação do motor.

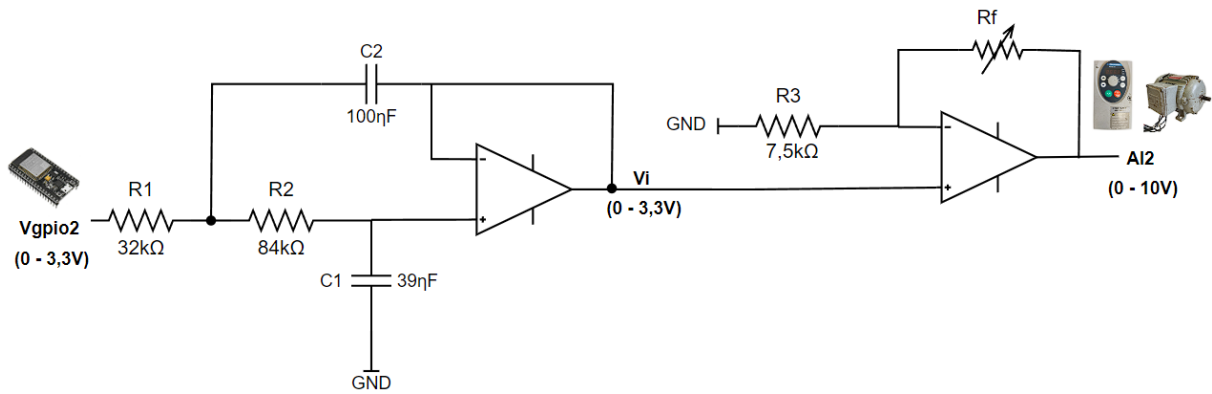
Essa alteração pode ser monitorada tanto pelo display do inversor quanto pela interface do servidor web, proporcionando um acompanhamento detalhado e prático do funcionamento do sistema.

Figura 52 - Bornes do controle de tensão analógica do inversor.



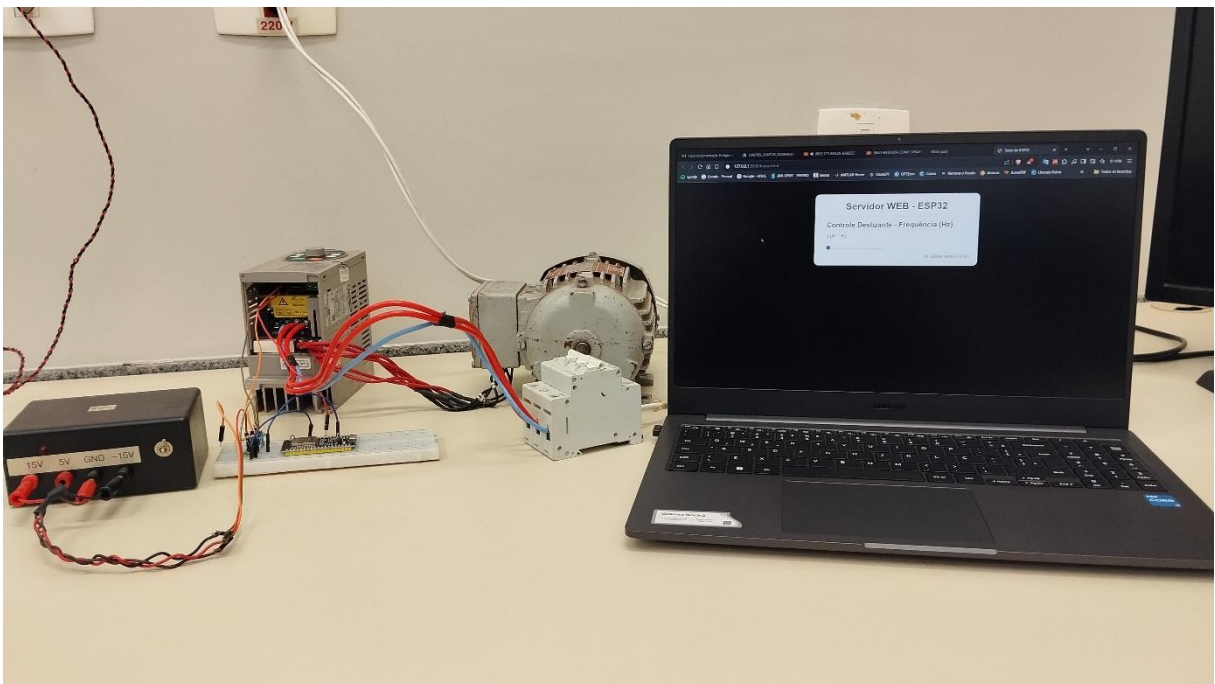
Fonte: (SCHNEIDER ELECTRIC, [s.d.]).

Figura 53 - Diagrama completo do sistema proposto.



Fonte: Próprio autor.

Figura 54 - Sistema completo montado no laboratório BATLAB.



Fonte: Próprio autor.

---

## 5 CONCLUSÕES

---

A priori, o projeto desenvolvido neste trabalho de conclusão revelou-se uma solução eficiente no âmbito da Internet das Coisas (IoT), reforçando a relevância do desenvolvimento de iniciativas que busquem soluções nessa área.

A proposta inicial consistiu no controle de velocidade de um motor de indução trifásico por meio de um controle deslizante de frequência, representado em uma interface web interativa. A implementação prática dessa funcionalidade foi viabilizada pelo microcontrolador ESP32, uma alternativa de baixo custo e fácil integração, programado na plataforma Arduino IDE.

Todos os equipamentos, componentes e circuitos montados para a execução do projeto funcionaram adequadamente, evidenciando a eficácia do desenvolvimento, especialmente considerando sua simplicidade e aplicabilidade prática.

Os resultados obtidos no controle de velocidade do motor de indução foram promissores. A interface web demonstrou boa precisão na exibição dos dados de frequência, quando comparados aos apresentados no display do inversor de frequência utilizado. Ademais, foi possível realizar o controle da velocidade a uma distância considerável do local onde o sistema estava instalado.

Como proposta para trabalhos futuros, sugere-se a inclusão de sensores de temperatura ou de ruído para a aquisição de dados do motor de indução, com a possibilidade de monitoramento remoto via interface web por meio de gráficos. Essa implementação permitiria a realização de manutenção preventiva no motor, aumentando sua vida útil e assegurando maior eficiência ao sistema.

---

## 6 REFERÊNCIAS

---

ALEX MARTINS DUARTE et al. **IOT EM MOTORES DE INDUÇÃO TRIFÁSICO**. São Bernardo do Campo - SP: [s.n.].

ALEXANDRE ABRAHAM PINTO DA CUNHA. **Sistema de Internet das Coisas para Monitoração de Motores Elétricos Industriais**. Brasília: [s.n.].

ANA LUÍSA FUNARI PIMENTA. **Projeto de filtros analógicos**. Cornélio Procópio - PR: [s.n.].

ANTONIO M. S. FRANCISCO. **Motores de indução trifásicos**. [s.l: s.n.].

ANTONIO PERTENCE JR. **Amplificadores Operacionais e Filtros Ativos**. 8ª ed. [s.l: s.n.].

ARDUINO. **Software Arduino IDE 2.3.3**. Disponível em: <<https://www.arduino.cc/en/software>>. Acesso em: 26 ago. 2024.

BLOG SMART KITS. **ESP32 pinout - Guia Básico de GPIOs**. Disponível em: <<https://blog.smartkits.com.br/esp32-pinout-guia-basico-de-gpios/>>. Acesso em: 20 nov. 2024.

BNDES. **Estudo “Internet das Coisas: um plano de ação para o Brasil”**. Disponível em: <<https://www.bndes.gov.br/wps/portal/site/home/conhecimento/pesquisaedados/estudos/estudo-internet-das-coisas-iot/estudo-internet-das-coisas-um-plano-de-acao-para-o-brasil>>. Acesso em: 13 out. 2024.

BNDES. **BNDES Pilotos IoT - Internet das Coisas**. Disponível em: <<https://www.bndes.gov.br/wps/portal/site/home/onde-atuamos/inovacao/internet-das-coisas>>.

---

coisas/bndes-projetos-piloto-internet-das-coisas/bndes-pilotos-iot-internet-das-coisas>.

Acesso em: 13 out. 2024.

CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU. **Fundamentos de Circuitos Elétricos**. 5ª ed. [s.l: s.n.].

CORDERO, R. et al. Educational Low-Cost Frequency Response Analyzer using Arduino and Software Octave. **2022 14th Seminar on Power Electronics and Control, SEPOC 2022**, 2022.

DANIEL CRUVINEL CORRAL; GUILHERME AFONSO MAZANTI. **Filtros Ativos**. São Carlos - SP: [s.n.].

DANIEL W. HART. **Eletrônica de Potência - análise e projetos de circuitos**. 1. ed. [s.l: s.n.].

DOKIC, K. Microcontrollers on the edge – is esp32 with camera ready for machine learning? **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 12119 LNCS, p. 213–220, 2020.

DR.P.B.PATHAK. Internet of Things: A Look at Paradigm Shifting Applications and Challenges. **International Journal of Advanced Research in Computer Science**, 2016.

EHUD RESHEF; CARLOS CORDEIRO. **Future Directions for Wi-Fi 8 and Beyond**.

[s.l: s.n.]. Disponível em:

<<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9864321&tag=1>>.

Acesso em: 14 nov. 2024.

---

ELOI, R.; FRANÇA, O.; CAMPOS, M. DE. **APLICAÇÃO DE IOT PARA ACIONAMENTO E MONITORAMENTO DE UM SISTEMA DE PARTIDA DE MOTORES ATRAVÉS DE MODULO RELÉ WI-FI**. [s.l: s.n.].

ESPRESSIF SYSTEMS. **ESP8266EX Datasheet**. [s.l: s.n.]. Disponível em: <<https://encurtador.com.br/ideB0>>. Acesso em: 29 out. 2024.

ESPRESSIF SYSTEMS. **ESP32 Series Datasheet Version 4.7**. [s.l: s.n.]. Disponível em: <<https://encurtador.com.br/Li2RW>>. Acesso em: 29 out. 2024.

FORBES. **IoT: até 2025, mais de 27 bilhões de dispositivos estarão conectados**. Disponível em: <<https://forbes.com.br/forbes-tech/2022/08/iot-ate-2025-mais-de-27-bilhoes-de-dispositivos-estara-conectados/>>. Acesso em: 21 out. 2024.

GOOGLE TRENDS. **Internet das coisas - Pesquisar - Google Trends**. Disponível em: <<https://trends.google.com.br/trends/explore?date=2004-01-01%202024-10-24&q=%2Fm%2F02vnd10&hl=pt>>. Acesso em: 22 out. 2024a.

GOOGLE TRENDS. **esp32 - Pesquisar - Google Trends**. Disponível em: <<https://trends.google.com.br/trends/explore?date=2016-09-06%202024-10-15&q=esp32&hl=pt>>. Acesso em: 14 out. 2024b.

INDIRA NIGHT. **Conectando o Arduino à web: Desenvolvimento de Frontend Usando JavaScript**. 1. ed. [s.l: s.n.].

MAREK BABIUCH; PETR FOLTÝNEK; PAVEL SMUTNÝ. **Using the ESP32 Microcontroller for Data Processing**. [s.l.] Institute of Electrical and Electronics Engineers, 2019.

MCKINSEY. **The next generation of connected IoT**. [s.l: s.n.].

---

MENEZES, H. F. **UMA REFLEXÃO SOBRE O HTML5: COMO ESSA TECNOLOGIA TEM POSSIBILITADO A CRIAÇÃO DE PÁGINAS WEB MAIS INTERATIVAS.** [s.l: s.n.].

PRESIDÊNCIA DA REPÚBLICA. **Decreto - Plano Nacional de Internet das Coisas.** [s.l: s.n.]. Disponível em: <[www.planalto.gov.br/ccivil\\_03/\\_Ato2019-2022/2019/Decreto/D9854.htm](http://www.planalto.gov.br/ccivil_03/_Ato2019-2022/2019/Decreto/D9854.htm)>.

RANDOM NERD TUTORIALS. **Getting Started with the ESP32 Development Board.** Disponível em: <<https://randomnerdtutorials.com/getting-started-with-esp32/#esp32-intro>>. Acesso em: 20 ago. 2024.

RFID JOURNAL. **That “Internet of Things” Thing.** Disponível em: <<https://www.rfidjournal.com/expert-views/that-internet-of-things-thing/73881/>>. Acesso em: 22 out. 2024.

RIOS, A. DE A. **Aplicação de internet das coisas no monitoramento de corrente, tensão e temperatura em motor de indução trifásico.** [s.l.] Universidade Federal de Uberlândia - UFU, jul. 2019.

ROBERT L. BOYLESTAD; LOUIS NASHELSKY. **Dispositivos Eletrônicos e Teoria dos Circuitos.** 11<sup>a</sup> ed. [s.l: s.n.].

SANTOS, B. P. et al. **Internet das Coisas: da Teoria à Prática.** Belo Horizonte, MG: [s.n.].

SCHNEIDER ELECTRIC. **Manual de operação e programação -Altivar 31.** [s.l: s.n.].

SHULONG WANG et al. **A Novel IoT Access Architecture for Vehicle Monitoring System.** Beijing, China: IEEE, 2016.

---

STEPHAN D. UMANS. **Máquinas Elétricas de Fitzgerald e Kingsley**. 7. ed. [s.l: s.n.].

STEPHEN J. CHAPMAN. **Fundamentos de Máquinas Elétricas**. 5. ed. [s.l: s.n.].

TEXAS INSTRUMENTS. **TL082 Wide Bandwidth Dual JFET Input Operational Amplifier**. [s.l: s.n.]. Disponível em: <[www.ti.com](http://www.ti.com)>.

TEXAS INSTRUMENTS. **Active Filter Design Techniques**. Dallas, Texas: [s.n.].

WEG. **GUIA DE ESPECIFICAÇÃO MOTORES ELÉTRICOS**. [s.l: s.n.].  
Disponível em: <[www.weg.net](http://www.weg.net)>.

WENDLING, M. **Amplificadores operacionais**. [s.l: s.n.].

---

## 7 APÊNDICES

---

### 7.1 PROGRAMAÇÃO NO ARDUINO IDE

```
/*Autor: Gabriel Santos Rodrigues
Adaptado de https://randomnerdtutorials.com/esp32-web-server-slider-pwm*/

// 1. Bibliotecas importadas
#include <WiFi.h>           // Gerencia as configurações do wi-
                             fi
#include <AsyncTCP.h>        // TCP assíncrono/ Manipulação de rede
                             assíncrona
#include <ESPAsyncWebServer.h> // Cria um servidor WEB Assíncrono

// 2. Definição de credenciais e variáveis
const char* ssid = "Redmi Note 12"; // Usuário da rede
const char* password = "gabriel01"; // Senha da rede

const int carga = 2;        // Pino de saída GPIO2
float sliderValue = 0.0;    // Armazena o valor inicial do controle deslizante
                             como float (frequência em Hz)

const char* PARAM_INPUT = "value"; // Parâmetro da solicitação HTTP

// 3. Criação do Servidor WEB
AsyncWebServer server(80); // Servidor configurado na porta padrão 80

// 4. HTML5, CSS3 e JavaScript
const char index_html[] PROGMEM = R"rawliteral(
<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <meta http-equiv='X-UA-Compatible' content='IE=edge'>
  <title>Teste do ESP32</title>
  <meta name='viewport' content='width=device-width, initial-scale=1'>
  <style>
    *{
      padding: 0;
      margin: 0;
      font-family: Arial, Helvetica, sans-serif;
```

## APÊNDICES e/ou ANEXOS

---

```

        color: #333;
        box-sizing: border-box;
    }
    body{
        background-color: #000;
        background-position: center;
        background-size: cover;
    }
    #pwmSlider{
        background: #fdfdfd;
        margin: 1rem;
        margin-top: 1.5rem;
        width: 200px;
    }
    .interfaceBranca{
        max-width: 500px;
        margin: 3rem auto;
        background-color: #fdfdfd;
        padding: 1.5rem;
        border-radius: 15px;
    }
    #titulo_page{
        text-align: center;
        padding: 0 1rem 1rem;
        border-bottom: 1px solid #ccc;
    }
    #texto_2, #textSliderValue{
        padding: 1rem;
    }
    .eu{
        text-align: end;
        font-size: .8rem;
    }
</style>
</head>
<body>
    <div class="interfaceBranca">
        <h1 id="titulo_page">Servidor WEB - ESP32</h1>
        <h2 id="texto_2">Controle Deslizante - Frequência (Hz)</h2>
        <p><span id="textSliderValue">%SLIDERVALUE%</span> Hz</p>
        <p><input type="range" onchange="updateSliderPWM(this)" id="pwmSlider"
min="0" max="255" value="%SLIDERVALUE%" step="1" class="slider"></p>
        <p class="eu">by Gabriel Santos (UFMS)</p>
    </div>

```

## APÊNDICES e/ou ANEXOS

---

```

    <script>
        function updateSliderPWM(element){
            var sliderValue = document.getElementById("pwmSlider").value;
            document.getElementById("textSliderValue").innerHTML =
(sliderValue * 60 / 255).toFixed(2); // Mostra a frequência calculada
            var xhr = new XMLHttpRequest();
            xhr.open("GET", "/slider?value=" + sliderValue, true);
            xhr.send();
        }
    </script>
</body>
</html>
)rawliteral";
// HTML armazenado na variável PROGMEM

// 5. PROCESSAMENTO: SUBSTITUI VALORES NO HTML
String processor(const String& var){
    if (var == "SLIDERVALUE"){
        return String((sliderValue * 60.0 / 255.0), 2); // Converte o valor do
controle deslizante em PWM(0-255) para Hz(0-60Hz), com 2 casas decimais
    }
    return String();
}

// 6. CONFIGURAÇÕES NO SETUP
void setup(){
    Serial.begin(115200); // Inicia a comunicação
    Serial/monitor serial

    WiFi.begin(ssid, password); // Conecta ao Wi-Fi
    while (WiFi.status() != WL_CONNECTED) {
        delay(1000);
        Serial.println("Connecting to WiFi..");
    }
    Serial.println(WiFi.localIP()); // Exibe o IP local do ESP32 no
monitor serial

    // Configura a página inicial
    server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
        request->send_P(200, "text/html", index_html, processor);
    });

    // Atualiza o valor do controle deslizante
    server.on("/slider", HTTP_GET, [] (AsyncWebServerRequest *request) {

```

## APÊNDICES e/ou ANEXOS

---

```

    String inputMessage;
    if (request->hasParam(PARAM_INPUT)) {
        inputMessage = request->getParam(PARAM_INPUT)->value();
        sliderValue = inputMessage.toFloat(); // Converte o valor do controle
        deslizante para float
        float freqValue = (sliderValue * 60.0) / 255.0; // Escala o valor de 0-
        255 para 0-60 Hz
        int pwmDutyCycle = map(sliderValue, 0, 255, 1, 255); // Mantém o ciclo
        de trabalho proporcional

        // Escreve diretamente no pino de carga usando analogWrite para ajustar
        o ciclo de trabalho
        analogWrite(carga, pwmDutyCycle);
    }
    Serial.println(inputMessage);
    request->send(200, "text/plain", "OK"); // Responde com "OK" a
    solicitação
    });

    server.begin(); // Inicia o servidor
}

// 7. Função Principal
void loop() {
}

```

## 7.2 PROGRAMAÇÃO NO MATLAB

```

%% Autor: Gabriel Santos Rodrigues
% Filtro PB de 2ª ordem
% Sallen-Key Butterworth

%% Dados iniciais

clc;
fc = 50;           % frequência de corte (Hz)
wc = 2*pi*fc;      % frequência de corte (rad/s)
A0 = 1;            % ganho linear unitário
a1 = 1.4142;        % coeficiente real de Butterworth 1 (sqrt(2))
b1 = 1.0000;        % coeficiente real de Butterworth 2
Q = sqrt(b1)/a1;    % Coef. de Qualidade

c1 = 39e-9;         % Capacitor 1 - 39nF
c2 = 100e-9;        % Capacitor 2 - 100nF

```

## APÊNDICES e/ou ANEXOS

---

```

c2_min = c1 * (4*b1) / (a1^2);    % Capacitor 2 mín - 78nF

%% Cálculos dos resistores R1 e R2

aux = sqrt((a1^2 * c2^2) - (4 * b1 * c1 * c2));
R1 = (a1 * c2 - aux) / (4 * pi * fc * c1 * c2);
R2 = (a1 * c2 + aux) / (4 * pi * fc * c1 * c2);

fprintf('Valor de R1: %.2f kΩ\n', R1/1000);
fprintf('Valor de R2: %.2f kΩ\n', R2/1000);

%% Função de Transferência

num = 1;
den = [(wc^2 * R1 * R2 * c1 * c2), wc * c1 * (R1 + R2), 1]; % numerador
H = tf(num, den); % denominador

%% Gráfico da resposta em frequência

figure(1)
bode(H)
grid on
title(['Filtro Passa-Baixa de 2ª Ordem de Butterworth']);

```