



**UNIVERSIDADE ZAMBEZE**  
**FACULDADE DE CIÊNCIAS E TECNOLOGIA**

Licenciatura em Engenharia Eléctrica

**Integração de IoT e Gêmeos Digitais para Optimização do Desempenho de Motores  
Eléctricos na Austral Cimento Sofala S.A**

Santos Arcanjo Bobone Ainda-Mais

Beira, Julho de 2025



**UNIVERSIDADE ZAMBEZE**  
**FACULDADE DE CIÊNCIAS E TECNOLOGIA**

Licenciatura em Engenharia Eléctrica

**Integração de IoT e Gêmeos Digitais para Optimização do Desempenho de Motores  
Eléctricos na Austral Cimento Sofala S.A**

Monografia apresentada ao Departamento de Engenharia Eletromecânica da Faculdade de Ciências e Tecnologia da Universidade Zambeze, como requisito parcial para a obtenção do grau de Licenciatura em Engenharia Eléctrica.

Santos Arcanjo Bobone Ainda-Mais

**Orientador:**

Eng.º. António Chama

Beira, Julho de 2025

## **Declaração de Honra**

Eu, **Santos Arcanjo Bobone Ainda-Mais**, declaro que esta monografia nunca foi apresentada para a obtenção de qualquer grau académico ou em qualquer outro âmbito, e que a mesma constitui o resultado do meu trabalho. Este trabalho é fruto do meu esforço e dedicação, realizado sob a orientação do meu supervisor, sendo o seu conteúdo original. As fontes consultadas estão devidamente referenciadas ao longo do texto e nas referências bibliográficas.

Esta monografia é apresentada em cumprimento parcial dos requisitos para a obtenção do grau de Licenciatura em Engenharia Eléctrica, pela Universidade Zambeze.

Beira, julho de 2025

---

(Santos Arcanjo Bobone Ainda-Mais)

## PARECER DO ORIENTADOR

O orientador **António Chama**, docente na Faculdade de Ciências e Tecnologia da Universidade Zambeze, vem na qualidade de orientador da monografia de licenciatura submetida por Santos Arcanjo Bobone, com o tema “**Integração de IoT e Gêmeos Digitais para Optimização do Desempenho de Motores Eléctricos na Austral Cimento Sofala S.A**” declarar que o trabalho apresentado pela candidata reúne a qualidade científica e as condições necessárias para que a estudante seja submetida a prova de defesa para obter o grau de licenciatura em engenharia eléctrica.

Faculdade de Ciências e Tecnologia da Universidade Zambeze, Julho de 2025

O orientador

---

(Eng.º António Chama)

## **Agradecimentos**

Honestamente, eu não sei como consegui chegar até aqui, mas com certeza não teria sido possível sem o apoio das pessoas que amo. Primeiramente, agradeço a Jeová Deus, por me ter dado força, sabedoria e fé ao longo deste percurso. Em cada desafio, senti Sua presença e direção, e sem isso nada seria possível. Agradeço profundamente aos meus pais, que sempre estiveram ao meu lado com palavras de apoio, sacrifício silencioso e amor incondicional. Foram eles os meus pilares, mesmo quando tudo parecia difícil. Este trabalho é também fruto do vosso esforço, dedicação e confiança em mim.

Aos meus colegas e grandes amigos Sivite Alberto, Jonass Salvador, Edvânia Hermenegildo, Edson José e Isacar Maria, o meu muito obrigado. O convívio, as conversas, as noites mal dormidas e as risadas tornaram a jornada mais leve. Cada um de vocês deixou uma marca única nesta caminhada. A faculdade não teria sido a mesma sem o vosso companheirismo e amizade. Aos professores e demais colegas do curso, que compartilharam comigo os mesmos desafios e aprendizados, deixo também o meu sincero reconhecimento. Como diria alguém muito sábio: "O verdadeiro curso são os amigos que fazemos ao longo do caminho." Por fim, a todos aqueles que me apoiaram direta ou indiretamente — seja com uma palavra de incentivo, uma ajuda técnica ou mesmo um gesto de amizade — deixo o meu mais profundo agradecimento. A todos vocês, com sinceridade e gratidão: Muito obrigado.

## **Dedicatória**

*Dedico este trabalho, primeiramente, à minha família, pelo amor incondicional e apoio constante durante toda a minha caminhada acadêmica. Aos meus amigos e colegas de faculdade, que estiveram ao meu lado nos momentos de dificuldade e aprendizado, deixo minha sincera gratidão pela partilha, companheirismo e incentivo. Mas, de forma muito especial, dedico este trabalho ao meu pai, Arcanjo Bobone Ainda-Mais. Seu exemplo de força, dedicação e integridade foi a minha maior inspiração ao longo deste percurso. Esta conquista é também sua.*

***“Procurei homens grandiosos,  
mas encontrei apenas «sombras»  
repetindo apenas ideais que não  
compreendem”***

***(Friedrich Nietzsche)***

## Índice:

<b>Lista de Figuras.....</b>	<b>VII</b>
<b>Lista de abreviações e siglas.....</b>	<b>IX</b>
<b>Lista de Tabelas.....</b>	<b>X</b>
<b>Resumo.....</b>	<b>XI</b>
<b>Abstract.....</b>	<b>XII</b>
<b>CAPÍTULO I: INTRODUÇÃO.....</b>	<b>1</b>
1.2 Justificativa.....	3
1.3 PROBLEMATIZAÇÃO .....	4
1.3.2 Hipótese .....	5
1.4 Objectivos .....	5
1.4.1 Geral.....	5
1.4.2 Especificos.....	6
1.5 Delimitação .....	6
1.6 Enquadramento.....	6
<b>CAPÍTULO II: REVISÃO BIBLIOGRÁFICA.....</b>	<b>7</b>
2.1 GÊMEOS DIGITAIS .....	7
2.2 Internet das Coisas .....	12
2.2.1 Segurança na Internet das Coisas .....	15
2.3 Motores Eléctricos Industrias.....	15
2.3.1 Manutenção Preditiva .....	19
<b>CAPÍTULO III: METODOLOGIA.....</b>	<b>20</b>
3.1 Maquinas Eléctricas Industrias.....	20
3.1.1 Motores Eléctricos .....	20
3.2 Inversor de frequência CFW-08 .....	23
3.3 Controladores e Atuadores .....	26
3.3.1 Microcontrolador ESP32 .....	26
3.3.2 Sensores de Temperatura, Ruído e Vibração.....	29
3.4 Softwares .....	34
3.4.1 Arduino IDE .....	35
3.4.2 ScadaBR – Sistema de Controlo e Aquisição de Dados.....	36

<b>CAPÍTULO IV: APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS.....</b>	<b>38</b>
4.1 Sistema de Monitorização de Variáveis .....	39
4.1.1 Alimentação e Ligação do Inversor de Frequência .....	40
4.1.2 Parametrização do Inversor CFW-08 .....	41
4.1.3 Conexão entre o ESP32 e os Sensores.....	42
4.1.4 Programação dos dispositivos no Arduino IDE .....	48
4.2 Sistema Supervisório.....	56
4.2.1 Visão Geral do Sistema Supervisório.....	57
4.2.2 Fontes de Dados (Data Sources).....	58
4.2.3 Sistema de Alarmes .....	66
4.2.4 Representações Gráficas.....	67
4.3 Sistema de Inteligência na Indústria 4.0: Aplicação com Motores Eléctricos .....	73
<b>CAPÍTULO V: CONCLUSÕES E SUGESTÕES.....</b>	<b>76</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>78</b>
<b>Apêndice A– Protocolo Modbus RTU.....</b>	<b>81</b>
<b>Apêndice B– Configuração do Conversor Serial na Rede Modbus RTU .....</b>	<b>85</b>
<b>Anexo A– Códigos de programação .....</b>	<b>86</b>



## **Lista de Figuras**

<b>Figura 1:</b> Gêmeo Digital. ....	8
<b>Figura 2:</b> Vertentes dos Gêmeos Digitais .....	9
<b>Figura 3:</b> Uso de Gêmeos Digitais na Saúde .....	11
<b>Figura 4:</b> Dimensões de comunicação e compartilhamento de informação na IoT.....	14
<b>Figura 5:</b> Visão em corte de um motor eléctrico .....	16
<b>Figura 6:</b> Princípio de funcionamento do motor eléctrico .....	17
<b>Figura 7:</b> Fluxograma de especificações para motores eléctricos.....	18
<b>Figura 8:</b> Uns do motor em estudo (Motor de Ventiladores).....	20
<b>Figura 9:</b> Motor trifásico para prototipagem .....	21
<b>Figura 10:</b> Chapa característica do motor da Lucas-Nuelle.....	22
<b>Figura 11:</b> Inversor de frequência CFW-08 .....	24
<b>Figura 12:</b> Módulo Conversor USB para RS-485 .....	25
<b>Figura 13:</b> Microcontrolador ESP32.....	27
<b>Figura 14:</b> Sensor de Temperatura DS18B20 .....	31
<b>Figura 15:</b> Módulo Sensor de Som KY-038 .....	33
<b>Figura 16:</b> Acelerômetro e Giroscópio MPU-6050 GY-521 .....	34
<b>Figura 17:</b> Bornes de potência do modelo utilizado do CFW-08 .....	40
<b>Figura 18:</b> Conexões de potência e aterramento .....	41
<b>Figura 19:</b> Pinagem do ESP32.....	44
<b>Figura 20:</b> Pinagem do DS18B20 .....	45
<b>Figura 21:</b> Pinagem do sensor de som .....	46
<b>Figura 22:</b> Esquema de Circuito Electrónico com ESP32 .....	48
<b>Figura 23:</b> Sistema de Aquisição de Dados .....	56
<b>Figura 24:</b> Data sources do projeto .....	58
<b>Figura 25:</b> Propriedades do data source ESP32 .....	61
<b>Figura 26:</b> Data points do ESP32.....	62
<b>Figura 27:</b> Propriedades inversor .....	63
<b>Figura 28:</b> Data points do Inversor .....	64
<b>Figura 29:</b> Data Source e Data Point do RPM .....	65

<b>Figura 30:</b> Tela Inicial.....	68
<b>Figura 31:</b> Tela do Motor.....	69
<b>Figura 32:</b> Tela de Parametrização .....	70
<b>Figura 33:</b> Tela de Alarme .....	71
<b>Figura 34:</b> Tela de Gráficos .....	72
<b>Figura 35:</b> Transmissão de dados modo RTU .....	83
<b>Figura 36:</b> Estrutura das mensagens .....	83
<b>Figura 37:</b> Gerenciador de Dispositivos do Windows .....	85

## Lista de abreviações e siglas

AI	Inteligência Artificial
ADC	Conversor Analógico-Digital (Analog-to-Digital Converter)
CFW-08	Inversor de Frequência da marca WEG, modelo CFW-08
CV	Cavalo-Vapor
dB	Decibel
DS18B20	Sensor Digital de Temperatura
ESP32	Microcontrolador de 32 bits com Wi-Fi e Bluetooth integrados
GPIO	Pino de Entrada/Saída Geral (General Purpose Input/Output)
IDE	Integrated Development Environment
I2C	Protocolo de Comunicação Serial (Inter-Integrated Circuit)
IoT	Internet das Coisas (Internet of Things)
KPI	Indicador-Chave de Desempenho (Key Performance Indicator)
KY-038	Sensor de Som com saída analógica e digital
Modbus RTU	Protocolo de Comunicação Serial Modbus (Remote Terminal Unit)
MQTT	Message Queuing Telemetry Transport
MPU-6050	Sensor de Movimento com Acelerômetro e Giroscópio integrados
Python	Linguagem de Programação de Alto Nível
RS-485	Padrão de Comunicação Serial utilizado em ambientes industriais
SCADA	Supervisory Control and Data Acquisition
SCADABR	Software Livre para Sistemas SCADA
TCP/IP	Transmission Control Protocol/Internet Protocol
USB	Universal Serial Bus
WEG	Empresa fabricante de motores e inversores elétricos

## Lista de Tabelas

<b>Tabela 1:</b> Funções Modbus disponíveis no CFW-08 .....	24
<b>Tabela 2:</b> Parâmetros da Comunicacao Serial .....	42
<b>Tabela 3:</b> Parâmetros de Regulação .....	42
<b>Tabela 4:</b> Pinagem do acelerômetro MPU-6050.....	47
<b>Tabela 5:</b> Tempos relacionados com a transferência de telegramas .....	84

## Resumo

O presente trabalho tem como objetivo apresentar o estudo realizado nas instalações da Austral Cimentos, mas o protótipo feito nas instalações da Universidade Zambeze, na província de Sofala, focando a aplicação da tecnologia de Gêmeos Digitais na monitorização e optimização de motores eléctricos industriais. A investigação abordou como essa tecnologia, integrada aos conceitos da Indústria 4.0, pode contribuir significativamente para a manutenção preditiva, previsão de falhas e melhoria do desempenho operacional dos motores críticos da fábrica. Inicialmente, é feita uma contextualização teórica sobre a relevância dos Gêmeos Digitais, a evolução tecnológica no sector industrial e o papel essencial dos motores eléctricos em processos contínuos de produção. A seguir, são apresentados os fundamentos técnicos adotados, bem como estudos relacionados que reforçam a importância académica e prática do tema. Partindo de uma necessidade real e de um contexto de inovação na planta industrial, o desenvolvimento do Gêmeo Digital envolveu a definição de componentes físicos e virtuais adequados, incluindo a construção de um sistema de aquisição de dados e outro de supervisão e controle. Foram utilizados o inversor de frequência WEG CFW-08, o microcontrolador ESP32, sensores de temperatura (DS18B20), ruído (KY-038) e vibração (MPU-6050), além de um conversor USB/Serial RS485. Esses dispositivos, integrados aos softwares Arduino IDE e ScadaBR, possibilitaram a criação de um sistema SCADA funcional para leitura, monitoramento e visualização em tempo real dos dados dos motores. A base deste modelo será posteriormente integrada a um sistema de inteligência computacional desenvolvido em ambiente de pesquisa de pós-graduação, permitindo o processamento distribuído, armazenamento em nuvem e análise preditiva dos dados. Por fim, avaliou-se a eficácia da solução proposta, os resultados obtidos e indicaram-se perspectivas futuras para aprimoramento e ampliação do uso de Gêmeos Digitais no ambiente industrial da Austral Cimentos.

**Palavras-Chaves:** Gêmeos Digitais. Motores Eléctricos, Manutenção Preditiva. Sistemas SCADA. Internet das Coisas.

## **Abstract**

This work presents a study conducted at the facilities of Austral Cimentos, with the prototype developed at University Zambeze, in Sofala Province, focusing on the application of Digital Twin technology for monitoring and optimizing industrial electric motors. The research investigates how this technology, aligned with Industry 4.0 concepts, can significantly contribute to predictive maintenance, fault forecasting, and operational performance improvement of critical factory motors. The study begins with a theoretical framework on the importance of Digital Twins, technological advancements in the industrial sector, and the fundamental role of electric motors in continuous production processes, especially in the cement industry. Next, technical foundations and related academic studies are presented, highlighting the growing relevance of the topic. From a real industrial need and innovation-driven context, the development of the Digital Twin involved defining suitable physical and virtual components, including the construction of a data acquisition system and a supervisory and control system. The hardware employed includes the WEG CFW-08 frequency inverter, ESP32 microcontroller, and DS18B20 (temperature), KY-038 (sound), and MPU-6050 (vibration) sensors, along with a USB/Serial RS485 converter. These components were integrated using Arduino IDE and ScadaBR software to develop a functional SCADA system for real-time data acquisition, monitoring, and visualization. The digital model was later enhanced with a computational intelligence system developed in a postgraduate research environment, allowing for distributed processing, cloud storage, and predictive analysis. Finally, the effectiveness of the proposed solution was evaluated, and future prospects for improving and expanding the use of Digital Twins in the industrial environment of Austral Cimentos were discussed.

**Keywords:** Digital Twins. Electric Motors. Predictive Maintenance. SCADA Systems. Internet of Things.

## CAPÍTULO I: INTRODUÇÃO

### 1. INTRODUÇÃO

#### 1.1 Contextualização

A Austral Cimentos Sofala S.A. (ACS) é uma empresa moçambicana localizada na província de Sofala, na zona industrial de Dondo, e iniciou as suas operações em 2014, tendo sido adquirida em 2016 pelo grupo alemão Heidelberg Materials. Com mais de 110 colaboradores, a ACS dedica-se à produção de cimento de alta qualidade, seguindo normas europeias (EN 197-1;2:2005/2006) e com certificação ISO 9001. A empresa tem como missão contribuir para o crescimento da economia moçambicana e para o desenvolvimento da infraestrutura nacional, investindo em tecnologia, segurança e qualificação profissional. Sob a direção de Hanns-Peter Mayer, profissional com mais de 30 anos de experiência na **HeidelbergCement**, a ACS destaca-se pela produção de cimento nas classes 32.5 N e 42.5 N, comercializado em sacos de 50 kg e a granel, destinado tanto ao mercado nacional como a países vizinhos como Malawi e Zimbabwe. A fábrica opera com um forte compromisso com a sustentabilidade, a segurança no local de trabalho e a ética empresarial, tendo sido reconhecida pelo Governo de Moçambique por suas boas práticas laborais. A maioria das matérias-primas são importadas através do Porto da Beira, embora haja um esforço constante para adquirir recursos localmente. A empresa mantém um elevado padrão de qualidade através de um laboratório interno e posiciona-se como uma referência no setor de cimento em Moçambique.

A Austral Cimentos Sofala esta dividida nessas áreas Principais:

- ✚ Seção de Preparação da Matéria-Prima- Composta por Britadores e alimentadores (até ao momento da visita, era composto por 10 motores electricos de pequeno porte), Transportadores de correia e elevadores de canecas.
- ✚ Moagem de Cimento e Ensacamento – Composta por:
  - Moinho de Cimento;
  - Separadores, transportadores e filtros;
  - Ensacadeiras e paletizadoras.

✚ Sistemas Auxiliares e Utilidades – Composta por:

- Sistemas de Ventilação, exaustão e bombas;
- Compressores de ar.

Para produzir o cimento são necessários vários processos e que nenhum desses processos deve ser interrompido, sob pena desse perder uma grande porção da matéria prima utilizada. Os motores elétricos são indispensáveis em qualquer fábrica de produção de cimentos, por essa razão o seu bom funcionamento é essencial para que não se perca a matéria prima. Para se obter um bom funcionamento dos motores trifásicos, a Austral Cimentos opta por realizar manutenções corretivas, preventivas e preditivas. As corretivas são realizadas caso ocorra uma falha no motor e precisa ser reparado ou substituído. Mas essa manutenção pode ocasionar paradas inesperadas que podem interromper a produção, custos elevados devido a solicitações de peças emergências, redução da vida útil do motor devido a fluxos recorrentes. As preditivas baseada em inspeções periódicas e ações programadas para evitar falhas. Neste caso faz se lubrificação periódica dos rolamentos para evitar desgaste, verificação de temperatura e vibração para detetar possíveis problemas, inspeção de ventilação e resfriamento para evitar superaquecimento e testes elétricos de isolamentos para evitar curto-circuitos reduzindo assim as falhas inesperadas e aumentando a vida útil dos motores.

As preditivas baseada no monitoramento contínuo de dados para prever falhas antes que aconteçam, fazendo análise de vibração monitoramento térmico (termografia infravermelha), análise da corrente elétrica. Permite programar paradas antes mesmo que ocorra uma falha, evita custos elevados. O projeto se baseará neste tipo de manutenção, fazendo o uso dos Gémeos digitais.

Até ao momento da visita esse tipo de manutenção era feito por meio de instalação de alguns sensores (**sensor Machine Sentry® MSF-2**), não havendo um sistema completo para o monitoramento e controlo dos motores indústrias trifásicas para a indentificação de falhas prévias. A inovação tecnológica é uma das principais forças motrizes do progresso econômico e social. O mercado de trabalho procura se reinventar diariamente em busca de novas soluções para um mercado mais produtivo e lucrativo, o que resulta em profundas transformações na sociedade, nas indústrias e na forma como nos relacionamos com as atividades do dia-a-dia. Partindo do ponto de vista de que o avanço da tecnologia é inevitável, as empresas precisam inovar para proteger sua



posição e ganhar mais espaço. Do contrário, elas são superadas por concorrentes com produtos melhores e processos mais eficientes. O mercado é altamente competitivo e é importante ter em mente que a inovação tecnológica é uma questão estratégica.

Independente do segmento de atuação das empresas, a transformação digital já não é mais novidade no mercado. Inteligência artificial, realidade aumentada, robótica, nuvem e Internet das Coisas (IoT). Termos que há alguns anos não eram nada conhecidos, hoje já fazem parte do cotidiano. Com a crescente adoção de tecnologias mais sofisticadas e inteligentes, a inovação na indústria não para de avançar. Esse fenômeno que está ocorrendo em larga escala nos últimos anos foi batizado de Indústria 4.0, ou também chamado de Quarta Revolução Industrial. Resumidamente, o enfoque são as fábricas inteligentes, as quais tornam os processos produtivos autônomos, eficientes e customizáveis.

Detalhadamente, a Indústria 4.0 é uma tendência mundial de digitalização da indústria, que visa integrar tecnologias avançadas, como inteligência artificial, Internet das Coisas, automação avançada e outras, para transformar a forma como a indústria é conduzida. Esta nova era da indústria busca aumentar a eficiência, flexibilidade e agilidade nas operações, melhorando a qualidade dos produtos e a eficiência dos processos. A Indústria 4.0 é considerada uma evolução da manufatura e é vista como uma oportunidade para aprimorar a competitividade global e criar novas formas de produção. (MARTINS, M. S.; PAULA; BOTELHO, 2021) Esse conceito surgiu para transformar a vida das pessoas, por meio do crescimento econômico, da geração de empregos qualificados e da elevação da qualidade de vida.

A presente monografia visa apresentar uma projeção de um sistema de internet das coisas através do conceito de Gêmeos Digitais como solução para manutenções sem falhas e eficiência energética.

## **1.2 Justificativa**

Com a utilização de tecnologias avançadas, é possível monitorar em tempo real a performance dos motores e identificar oportunidades de melhoria. Especificamente, podemos utilizar o conceito de Gêmeos Digitais, que são modelos virtuais de motores elétricos que permitem simular e monitorar suas operações de forma eficiente. Somado a isso, a integração de sensores e sistemas de controle permite ajustar automaticamente as configurações dos motores para otimizar sua eficiência e evitar

falhas. A utilização de um sistema SCADA permite monitorar, supervisionar e controlar as informações obtidas para facilitar o gerenciamento dos dados e tornar o processo intuitivo. Baseado nestas ideias que o presente trabalho será apresentado.

## **1.3 PROBLEMATIZAÇÃO**

### **1.3.1 Problema**


O sector industrial, especialmente a produção de cimento, demanda um controle rigoroso sobre seus processos produtivos para garantir eficiência energética, redução de custos operacionais e manutenção preditiva eficaz. Motores elétricos desempenham um papel crítico nesse ambiente, pois são responsáveis por movimentar equipamentos essenciais, como britadores, fornos rotativos, moinhos e transportadores de correia. No entanto, a falta de um sistema eficiente de monitoramento e controle pode resultar em falhas inesperadas, consumo excessivo de energia e tempo de inatividade não planejado, impactando diretamente a produtividade e a competitividade da fábrica.

Com os avanços da Indústria 4.0, a Internet das Coisas (IoT) surge como uma solução promissora para otimizar a gestão desses motores elétricos industriais. Sensores conectados a uma rede IoT podem coletar dados em tempo real sobre variáveis como temperatura, vibração, corrente elétrica e eficiência energética, permitindo diagnósticos mais precisos e ações preventivas antes que ocorram falhas críticas. Além disso, a integração desses dados com algoritmos de inteligência artificial pode potencializar a automação do controle dos motores, otimizando o consumo de energia e aumentando a vida útil dos equipamentos.

Apesar dos benefícios potenciais, a implementação de um sistema IoT em fábricas de cimento enfrenta desafios significativos, como a conectividade em ambientes industriais adversos, o tratamento e análise eficiente dos grandes volumes de dados gerados e a compatibilidade com sistemas legados já em operação. Além disso, é fundamental avaliar o custo-benefício dessa tecnologia para assegurar que a adoção do IoT traga vantagens econômicas e operacionais para a fábrica.


Diante desse cenário, surge a necessidade de investigar como um sistema de IoT pode ser eficaz na monitoração e controle de motores elétricos industriais, identificando as melhores práticas para sua implementação e os benefícios concretos que pode proporcionar à indústria cimenteira.

**Face aos pressupostos apresentados acima, questiona-se o seguinte?**

 **Como a Integração de IoT e Gêmeos Digitais pode otimizar a monitoração de motores elétricos industriais na Austral Cimentos, promovendo maior eficiência energética e redução de falhas operacionais?**

### 1.3.2 Hipótese

Segundo (SÉRGIO et al, 2017), actualmente, a busca por soluções que consigam trazer novos recursos financeiros para as empresas/ ou instituições, que não sejam agressivas ao meio ambiente e consigam impulsionar o mercado através de desenvolvimento de novos produtos e serviços, torna-se um desafio diário para todos os que estão envolvidos na gestão de empresas de energia, bem como aos responsáveis por trazerem inovações que atendam a estes anseios. A implementação de um sistema de Internet das Coisas (IoT) para a monitoração e controle de motores elétricos industriais na produção de cimento permitirá a coleta e análise contínua de dados operacionais, resultando na otimização da eficiência energética e na redução de falhas operacionais.

 **Melhoria na Manutenção Preditiva – O monitoramento em tempo real de variáveis como temperatura, vibração e corrente elétrica possibilitará a identificação precoce de anomalias, reduzindo paradas não planejadas e aumentando a vida útil dos motores.**

A adoção do IoT e gémeos digitais pode representar uma solução viável e vantajosa para a indústria, trazendo benefícios operacionais e económicos significativos.

## 1.4 Objectivos

### 1.4.1 Geral

Desenvolver um sistema de monitoramento inteligente de motores eléctricos industriais baseado em tecnologias de Internet das Coisas (IoT) e Gémeos Digitais.

### **1.4.2 Específicos**

- ✚ Revisão Bibliográfica
- ✚ Identificar os parâmetros operacionais críticos dos motores eléctricos
- ✚ Desenvolver um sistema IoT capaz de recolher, processar e transmitir os dados dos motores em tempo real
- ✚ Implementar uma interface gráfica interactiva
- ✚ Testar o sistema desenvolvido num ambiente simulado

### **1.5 Delimitação**

Este estudo focará na Projecção de um sistema de Internet das Coisas (IoT) para monitoramento de motores eléctricos industriais no processo de produção de cimento. A pesquisa se concentrará no monitoramento e controle de motores eléctricos responsáveis por processos críticos, como acionamento de britadores, moinhos, fornos rotativos e transportadores de correia. Motores de pequeno porte ou que não impactam diretamente na eficiência energética global da fábrica não serão analisados. Serão considerados sensores de monitoramento para variáveis como temperatura, vibração, corrente eléctrica e consumo energético, bem como a comunicação de dados via protocolos industriais seguros. Além disso, o estudo avaliará o uso de inteligência artificial para análise preditiva e otimização do desempenho dos motores.

### **1.6 Enquadramento**

Este tema enquadra-se nos objectivos do plano curricular do Curso de Licenciatura em Engenharia Eléctrica, administrado pela Faculdade de Ciências e Tecnologias da Universidade Zambeze – Beira, concretamente nas disciplinas de Controlo Automático, Sistemas Digitais, Instalações Eléctricas, Electrónica de Potencia, Tecnologia de Construção e Manutenção, Accionamentos Eléctricos, Medidas Eléctricas, Engenharia e Sociedade, Protecção e Automação de Sistemas Eléctricos, Maquinas Eléctricas, e entre outra.

## CAPÍTULO II: REVISÃO BIBLIOGRÁFICA

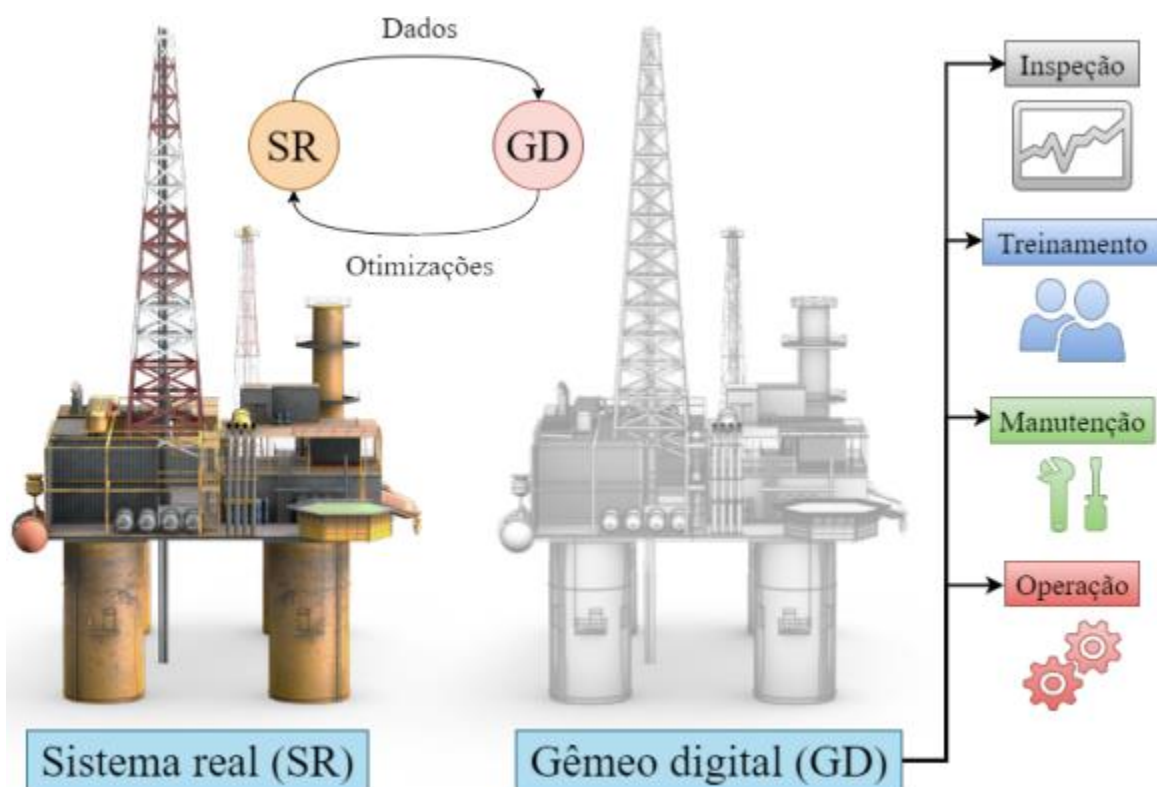
### *2.1 Gêmios Digitais*

Um gêmeo digital é uma representação digital de um ativo ou ambiente físico, como um carro, uma ponte ou um edifício. Pense nele menos como um modelo 3D estático tradicional e mais como um modelo de informações em constante mudança. É uma referência de dados comum criada durante a fase de planejamento de um projeto e abrange todas as fases do ciclo de vida de um ativo, do projeto à fabricação, da construção à operação e manutenção até seu uso ou reutilização futura.

Ao contrário dos modelos de dados estáticos, os gêmeos digitais são entidades dinâmicas e "vivas" que evoluem em tempo real. Eles registram, aprendem, atualizam e se comunicam continuamente com suas contrapartes físicas, trocando dados ao longo do ciclo de vida do ativo por meio de tecnologias de inteligência artificial (IA), aprendizado de máquina e Internet das Coisas (IoT). ( <https://www.autodesk.com/design-make/articles/what-is-a-digital-twin>). Os gêmeos digitais são projetados para fornecer uma compreensão detalhada de como um objecto ou sistema opera e podem ser usados para monitorar, analisar e otimizar o desempenho de sistemas e processos físicos (ATTARAN; CELIK, 2023). Como resultado, a implementação de gêmeos digitais levou a uma maior eficiência, segurança e sustentabilidade em vários setores. Eles podem replicar muitos itens, desde peças únicas de equipamento em uma fábrica a instalações completas, como turbinas eólicas e até cidades inteiras. Eles também podem ajudar as empresas a criar planos de negócios de longo prazo, identificar novas invenções e até simular cenários muito demorados ou caros para serem testados com objetos físicos.

A capacidade dos gêmeos digitais de se integrar entre os espaços físico e digital permite que o sistema físico ajuste seu comportamento em tempo real com base no feedback fornecido pelo gêmeo digital. Em outras palavras, a tecnologia de gêmeos digitais cria uma ponte entre os espaços físico e digital, permitindo que o gêmeo digital imite com precisão o estado do mundo real do gêmeo físico. Para criar essa ponte, é necessário desenvolver um forte vínculo entre os gêmeos físicos e digitais, o que pode ser alcançado pela circulação de dados sensoriais online entre eles e só pode ser desenvolvido depois de replicar com precisão o ambiente dinâmico do sistema físico e todos os fatores que impactam as operações desse sistema no espaço digital. Este processo, por

outro lado, necessita de outras tecnologias aliadas aos gêmeos digitais para ser desempenhado corretamente.

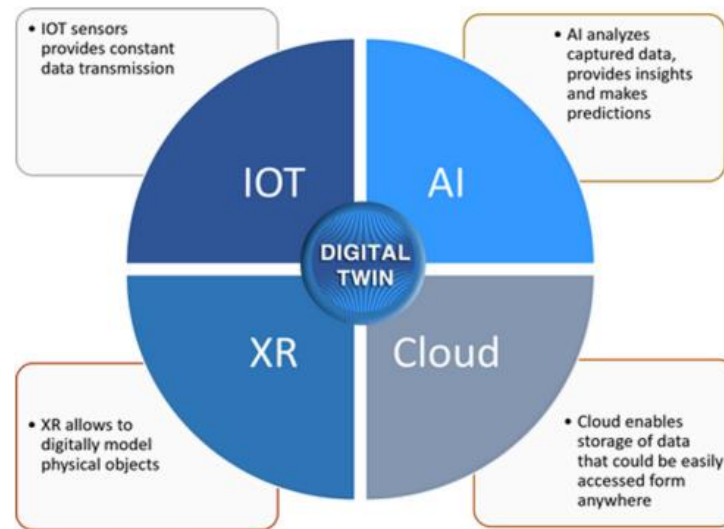


**Figura 1:** Gêmeo Digital.

**Fonte:** Dissertação\_2022097

De forma resumida, a tecnologia de gêmeos digitais se baseia em 4 outras vertentes tecnológicas para captura, rearmazenar dados em tempo real, analisar o valor das informações obtidas e criar representações virtuais coexistentes e interativas com seus objetos físicos: Internet das Coisas (IoT), Inteligência Artificial (AI), Realidade Estendida (XR) e Nuvem. A IoT é responsável pela conexão de uma rede de dispositivos conectados via internet que trocam dados entre si. Dessa forma, diferentes sensores, softwares e equipamentos comunicam-se para prover dados constantes e confiáveis para as aplicações de Gêmeos Digitais. Para construir seus modelos virtuais, a Inteligência Artificial pode auxiliar a analisar de forma avançada os dados obtidos, determinando valores e prioridades para cada dado, além de prover previsões e sugestões a partir das condições atuais do sistema. Enquanto que a Realidade Estendida é um termo genérico usado

para descrever tecnologias que mesclam os mundos físico e virtual como Realidade Virtual (VR), Realidade Aumentada (AR) e Realidade Mista (MR). Ela é utilizada para criar os modelos digitais e permitir a interação com o conteúdo digital. Por último, a Nuvem que guarda todas as informações obtidas e geradas em larga escala em um local seguro e de fácil acesso em qualquer lugar, reduzindo no processo efetivamente o tempo de computação de sistemas complexos (ATTARAN; CELIK, 2025).



**Figura 2:** Vertentes dos Gêmeos Digitais

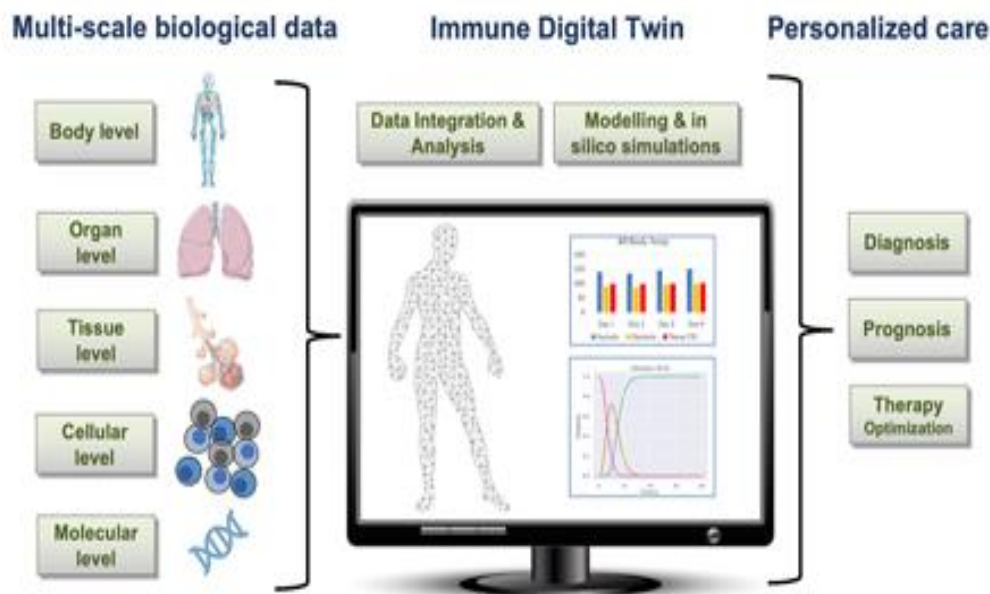
**Fonte:** (ATTARAN; CELIK, 2025)

A tecnologia de gêmeos digitais ganhou força significativa em uma ampla gama de sectores nos últimos anos. Dentre eles, podemos citar as áreas da saúde, aeroespacial, robótica, arquitectura, transporte, energia (ATTARAN; CELIK, 2023). Especificamente, na indústria da manufatura, por exemplo, os gêmeos digitais estão sendo usados para otimizar os processos de produção e melhorar a qualidade do produto. Em um estudo, o foco da pesquisa recaiu sobre a geração de um substituto digital de uma fábrica baseada em objectos reais em um ambiente construído com planos e documentação inexistentes ou insuficientes e com cenas complicadas (por exemplo, ambientes internos com objectos repetitivos de formato irregular dados de medição ruidosos como entrada) para melhorar os processos de produção e logística, levando à redução de custos. Os resultados mostram que mais de 80% dos objectos da fábrica puderam ser reconhecidos no modelo gerado com diferentes configurações arquitetónicas para otimizar a fábrica a depender dos requisitos (SOMMER et al., 2023).

Na indústria aeroespacial, os gêmeos digitais estão sendo usados para melhorar o design e o desempenho das aeronaves. Eles podem simular a aerodinâmica de uma aeronave e otimizar seu desempenho. Por exemplo, o estudo de um caso industrial derivado da colaboração com três fabricantes envolvidos no desenvolvimento de um sistema de propulsão eléctrica (EPSs) baseado na tecnologia Hall Effect Thruster (HET), propõe usar ativamente gêmeos digitais nas fases iniciais, onde o design pode ser amplamente alterado (PANAROTTO; ISAKSSON; VIAL, 2023). O estudo realiza uma abordagem econômica para identificar um conjunto de módulos gêmeos digitais confiáveis e robustos que podem ser reutilizados e recompostos para criar variantes de gêmeos digitais, seguindo uma abordagem de plataforma modular. Esses resultados apontam para o custo-benefício da aplicação de uma abordagem modular à criação de gêmeos digitais, como um meio de reutilizar os resultados dos testes físicos para validar novos projectos e seus intervalos de validade, podendo ser eficaz e útil nos estágios iniciais em que o design pode ser amplamente alterado, especialmente no contexto de sistemas avançados, como produtos aeroespaciais e espaciais.

Na área da saúde, os gêmeos digitais estão sendo usados para personalizar tratamentos médicos e melhorar os resultados dos pacientes. Eles podem simular o comportamento do corpo de um paciente e prever como ele responderá a diferentes tratamentos. Um esboço de alto nível de um roteiro para a construção de um gêmeo digital de uma pessoa é representado na Fig. 2.5. Para criar um modelo digital que possa ser testado e simulado, é preciso reunir enormes volumes de dados de saúde por meio de vários dispositivos IoT e usar modelos baseados em Inteligência Artificial. Com base no estilo de vida do paciente, hábitos alimentares regulares e dados de açúcar no sangue, essa tecnologia ajuda a alertar o paciente sobre prescrições, ajustes dietéticos, consultas médicas e outras situações. O gêmeo digital do próprio paciente usa insights de dados anteriores para ajudar a seleccionar o medicamento mais apropriado, prever os resultados de uma cirurgia específica e controlar doenças crônicas (HALEEM et al., 2023).





**Figura 3:** *Uso de Gêmeos Digitais na Saúde*

**Fonte:** <https://rdcu.be/c9uHR>, em 05/05/2025

Da mesma forma, no sector de energia, os gêmeos digitais estão sendo usados para otimizar o desempenho de usinas eléctricas e sistemas de energia renovável. Eles surgem como uma técnica potencial para melhorar o desempenho, reduzir os custos de manutenção e operação e garantir uma operação mais segura para qualquer sistema associado (SEMERARO et al., 2023). Na área de transporte, os gêmeos digitais estão sendo usados para na simulação de tráfegos de rodovias e sistemas de transporte. Utilizando softwares específicos somados a dados de tráfego em tempo real recém-disponíveis recebidos a cada minuto dos contadores de tráfego na rodovia de Genebra via ODPMS, a pesquisa realizada por (KUŠIĆ; SCHUMANN; IVANJKO, 2023) apresenta e testa pela primeira vez uma réplica digital de autoestrada ao vivo no simulador microscópico SUMO, refletindo com precisão a dinâmica real do tráfego com uma resolução temporal muito fina durante o tempo de execução do sistema. Este sistema, forma a base para o desenvolvimento da análise preditiva em tempo real como suporte para decisões críticas de segurança no gerenciamento de tráfego. Espera-se que o uso de gêmeos digitais desempenhe um papel crítico no desenvolvimento de cidades inteligentes. É importante destacar que, em todas as áreas citadas, a manutenção preditiva baseada em gêmeos digitais se faz presente e é um foco de pesquisa nos respectivos

campos de estudo (ZHONG et al., 2023). Diante da falha de um equipamento, a manutenção preditiva tradicional enfatiza a análise do estado do equipamento, a previsão de falha do equipamento e a tomada de decisão auxiliar. Essas funções não consideram a interação complexa e o mecanismo de evolução dos atributos estruturais e físicos no equipamento, e não podem considerar os parâmetros do equipamento em tempo real durante o processamento de dados. A manutenção preditiva tradicional só pode descrever o processo dinâmico sob uma certa escala espacial e de tempo, e não pode realizar manutenção dinâmica de longo prazo. A tecnologia de gêmeos digitais simula quando o equipamento está funcionando, coleta os dados do equipamento que mudam dinamicamente com o tempo e pode fornecer um serviço mais inteligente do que a manutenção preditiva tradicional, visto que pode realizar a manutenção dinâmica de longo prazo do equipamento.

É possível perceber como o uso de gêmeos digitais está se tornando cada vez mais popular em vários sectores. Existem infinitas outras aplicações que poderiam ser citadas e inclusive existem estudos somente para analisar artigos que mencionam o uso de gêmeos digitais afim de caracterizá-lo, identificando lacunas no conhecimento e áreas necessárias para pesquisas futuras (JONES et al., 2020). Os gêmeos digitais têm o potencial de revolucionar a maneira como projetamos, construímos e operamos sistemas físicos. Com sua capacidade de simular o comportamento e o desempenho de sistemas físicos em tempo real, espera-se que os gêmeos digitais continuem a desempenhar um papel significativo na melhoria da performance, segurança, manutenção preditiva, redução de custos, monitoração remota, aceleração do tempo de produção e sustentabilidade em todos os sectores.

Na prática, o Gémeo Digital do motor eléctrico é um modelo virtual tridimensional, integrado com dados reais recolhidos por sensores instalados no equipamento físico. Este modelo é continuamente atualizado com dados em tempo real (ou históricos), permitindo simular, prever e otimizar o comportamento e o desempenho do motor.

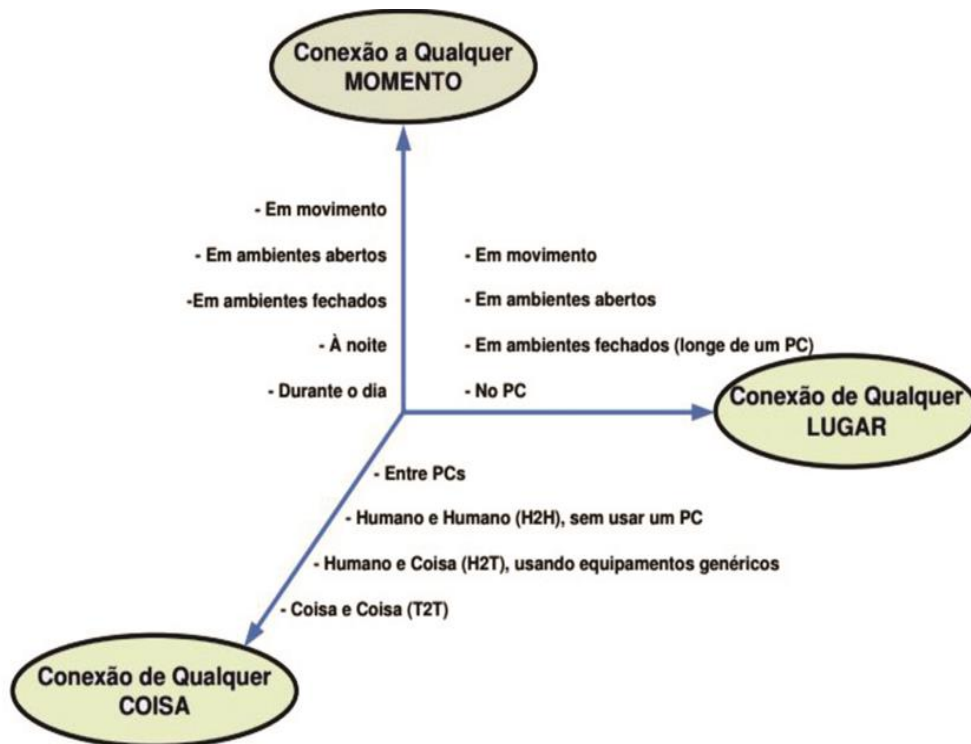
## ***2.2 Internet das Coisas***

A internet das coisas, ou IoT, é uma rede de dispositivos inter-relacionados que se conectam e trocam dados com outros dispositivos de IoT e com a nuvem. Os dispositivos de IoT

normalmente são equipados com tecnologias como sensores e software, e podem incluir máquinas mecânicas e digitais, além de objectos de consumo. Esses dispositivos abrangem tudo, desde itens domésticos do dia a dia até ferramentas industriais complexas. Cada vez mais, organizações de diversos sectores estão usando a IoT para operar com mais eficiência, oferecer um atendimento aprimorado ao cliente, aprimorar a tomada de decisões e aumentar o valor do negócio. Com a atual realidade tecnológica observou-se que houve um crescimento enorme das tecnologias de informação e comunicação. Com o passar dos dias as pessoas tendem a ficar mais dependentes de dispositivos conectados a Internet, como smartphones, tablets e notebooks, que são utilizados em atividades cotidianas nas mais diversas áreas profissionais e de lazer.

A rede mundial de computadores é o maior sistema de engenharia criado pelo homem (KUROSE; ROSS, 2010). A disponibilidade de acesso a Internet por dispositivos móveis, acarretou em uma rápida popularização nas últimas décadas (SANTO, 2015). Cenário favorável para inovações, ideias, teorias, como também divulgação e distribuição de novas tecnologias, entre elas, a Internet das Coisas (**IoT- Internet of Things**). O pesquisador do MIT (Massachusetts Institute of Technology) Kevin Ashton, utilizou-se do termo Internet das Coisas pela primeira vez, em 1999, em uma apresentação direcionada a empresa Procter & Gamble (SERAFIM, 2014). A IoT em uma definição mais abrangente, se relaciona com uma infraestrutura de hardware, software e serviços que têm o objetivo de conectar os mais diferentes objetos físicos, chamados de coisas, a rede de computadores (NETO, 2015). A IoT permite que os objetos conectados possuam a capacidade de disponibilizarem informações sobre sua atuação, proporcionando a eles inteligência e permitindo um melhor controle e monitoramento, disponibilizando notificações de alterações em seu estado (FERREIRA, 2014).

Na Figura 4, é possível observar três dimensões ligadas a IoT: lugar, momento e coisa. Estas dimensões incentivam e influenciam novas oportunidades, exemplos de mobilidade em diferentes situações de infraestrutura de rede. Isso torna a IoT uma tecnologia presente em toda parte.



**Figura 4:** Dimensões de comunicação e compartilhamento de informação na IoT

**Fonte:** Neto (2015) apud Pena-Lopez (2005)

Diante das variedades de aplicações e dos diversos segmentos de mercado, não se pode medir o verdadeiro potencial da IoT, pois ainda se encontra em fase inicial.

Segundo Atzori et al. (2010), existe um padrão da IoT, resultante do cruzamento de três visões diferentes: (a) orientada a coisas, (b) orientada à Internet e (c) orientada à semântica. A visão orientada das coisas determina meios que asseguram o melhor aproveitamento, de toda funcionalidade dos dispositivos de comunicação; a visão orientada e semântica, tem um foco na representatividade da informação gerada, juntamente com o armazenamento e pesquisa. Esta visão procura soluções que irão permitir um adequado tratamento para os dados que são produzidos pelos objetos; por fim, a visão orientada a internet, tem o objetivo de fornecer modelos e técnicas para uma comunicação transparente entre os dispositivos da rede.

De acordo com Lacerda (2015) o uso da Internet das Coisas pode trazer grandes benefícios para a sociedade, uma vez que se pode associá-la a muitos objetos do dia a dia, podendo atingir efeitos significativos em áreas como: segurança pessoal, conforto, lazer, meio ambiente, saúde.

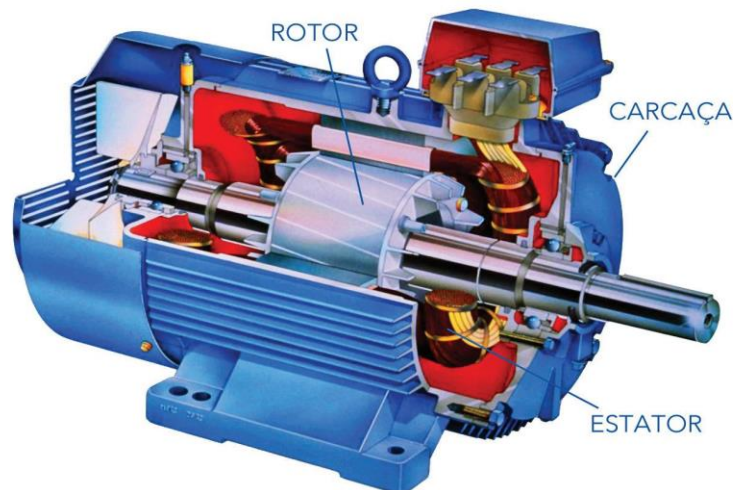
### ***2.2.1 Segurança na Internet das Coisas***

Um grande factor que ainda pesa na decisão da usabilidade desta tecnologia, é a segurança da conexão, pois os usuários da Internet já estão em constantes ataques que exploram as pequenas brechas. E como a maioria dos dispositivos que compõem a IoT, possuem funções e recursos limitados como o processamento, memória e energia, são mais suscetíveis a ataques. "O grande desafio é impedir o crescimento de tais ameaças ou pelo menos mitigar o seu impacto"(NETO, 2015, p. 25).

### ***2.3 Motores Eléctricos Indústrias***

Motores são dispositivos com papel imprescindível para a sociedade, amplamente utilizados em uma variedade de aplicações domésticas e industriais. Desde os primórdios da humanidade, o homem utiliza fontes motoras para realizar trabalho e desenvolver-se, seja na produção rural das primeiras comunidades agrícolas para alimentar-se e realizar comércio ou até em espaçonaves nos tempos modernos para ir ao espaço. Dentre os diferentes tipos de motores existentes (a combustão, ar comprimido), o motor eléctrico é o mais utilizado nos equipamentos em geral devido à versatilidade de adaptação a diferentes cargas somado às vantagens da energia eléctrica, por ser mais fácil de manusear, ecológica e barata. Eles são dispositivos que convertem energia eléctrica em energia mecânica e é possível encontrar este tipo de motor no cotidiano de uma pessoa através de equipamentos como portões electrónicos, elevadores, ventiladores, escadas rolantes e até em suas residências nos electrodomésticos: geladeiras, micro-ondas, liquidificadores, aspiradores, máquinas de lavar.

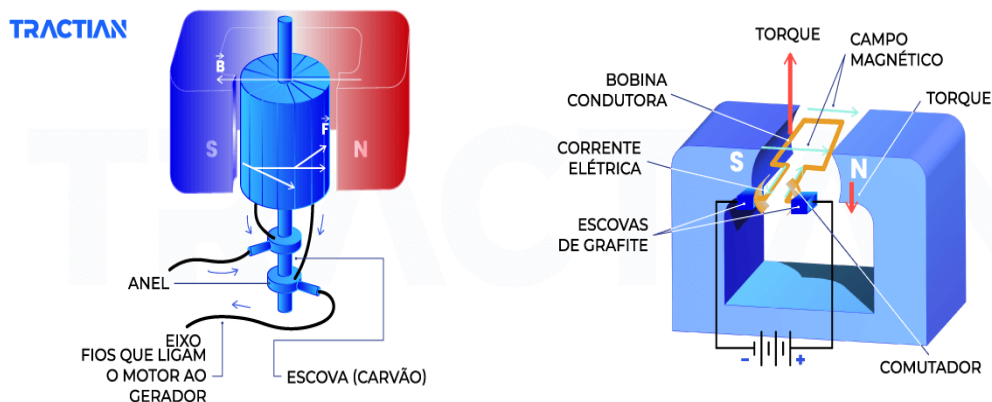
Em relação à indústria, motores eléctricos são componentes fundamentais, pois são usados para alimentar uma ampla gama de equipamentos e máquinas como: compressores, bombas, veículos eléctricos, misturadores, discos de corte. Eles fornecem a força motriz necessária para a produção de bens e serviços em vários sectores industriais, incluindo alimentos, químicos, têxtil, papel e celulose, mineração, entre outros. Em países industrializados, por exemplo, 65% do consumo de eletricidade das indústrias são responsáveis pelos motores eléctricos, e 90% deste consumo é em função dos motores de indução com uma configuração específica chamada de gaiola de esquilo (MARUTHI; PANDURANGA VITTAL, 2005).



**Figura 5:** Visão em corte de um motor eléctrico

**Fonte:** [https://issuu.com/sobratema/docs/mt\\_248\\_alta\\_sem\\_marca\\_de\\_corte\\_final/s/11158369](https://issuu.com/sobratema/docs/mt_248_alta_sem_marca_de_corte_final/s/11158369)

O motor eléctrico é dividido resumidamente em três partes principais: a carcaça, o estator e o rotor (Fig. 5). A carcaça é responsável por proteger as peças internas e dissipar o calor que é gerado pelo trabalho do motor. Já o estator é a parte estática fixa na carcaça e possui a função de fornecer o fluxo magnético que vai interagir com as partes girantes. Por fim, o rotor é a parte que gira do motor. Ele sofre influência de todo o processo eletromagnético e o transforma em movimento mecânico. O modo de funcionamento básico de um motor eléctrico é baseado na interacção entre o campo magnético gerado pelo fluxo de corrente eléctrica através dos enrolamentos do rotor e o campo magnético gerado pelo ímã permanente ou pela corrente eléctrica através dos enrolamentos de excitação do estator. As forças de atracção ou repulsão geram torque e consequentemente puxam ou empurram os polos móveis, definindo dessa forma a velocidade de giro a partir da intensidade do campo magnético. Este, por sua vez, é controlado pela frequência da corrente eléctrica que pode ser ajustada por meio de um inversor de frequência (FRANCHI, s.d.). De forma ilustrativa e com mais detalhes, a Fig. 6 explicita o princípio de funcionamento do motor eléctrico.

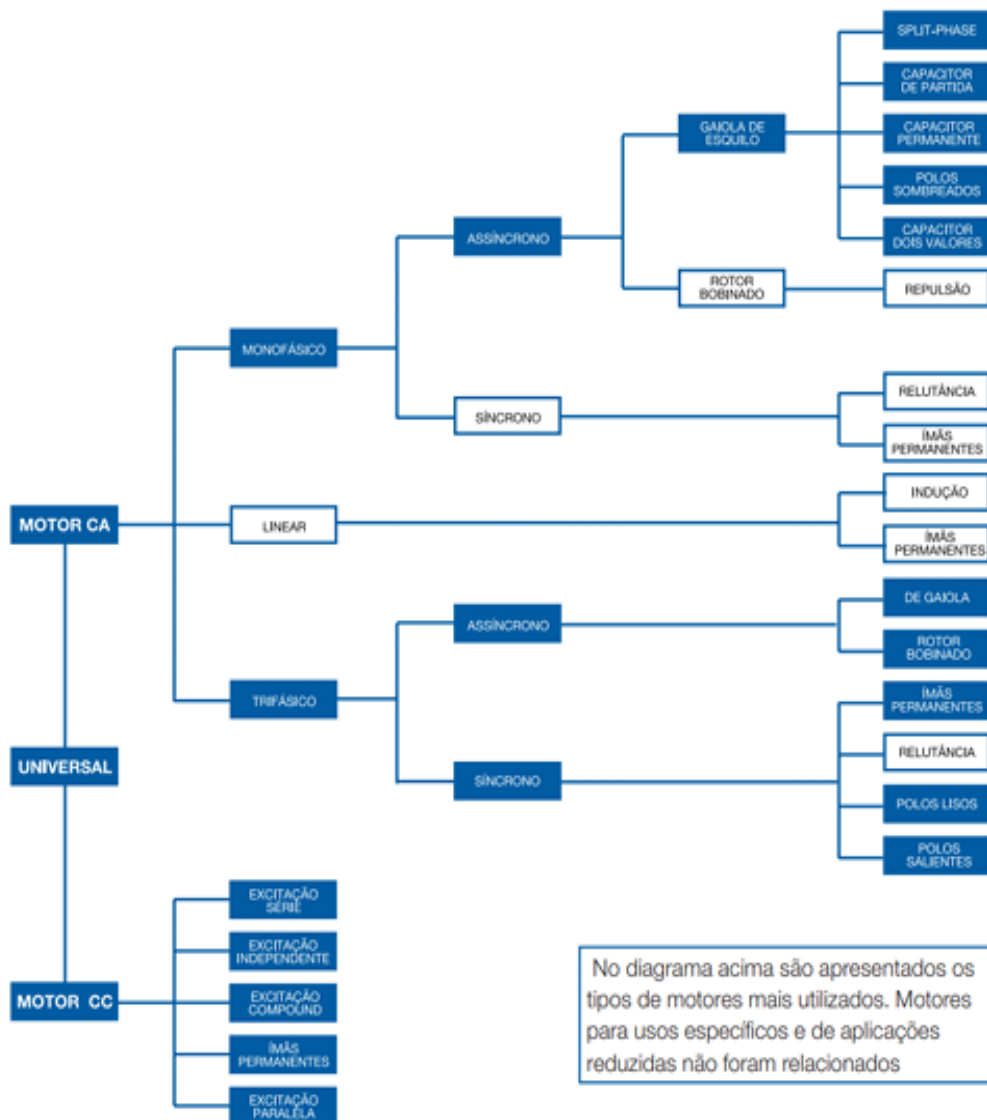


**Figura 6:** Princípio de funcionamento do motor elétrico

**Fonte:** <https://traction.com/blog/motores-eletricos-entenda-a-funcionalidade-desse-ativo>

Uma característica importante dos motores elétricos é a sua eficiência, visto que tem impacto directo na qualidade dos produtos, na segurança dos trabalhadores e na eficiência energética da planta. Uma melhora nela pode resultar em redução de custos operacionais, aumento da vida útil dos equipamentos e até redução de emissões de CO<sub>2</sub>. Para isso, existem vários tipos de motores eléctricos, os quais podem ser classificados como motores de corrente contínua (CC), corrente alternada (CA) ou motores universais (CC/CA). A partir destas categorias derivam-se outros tipos de motores em função de suas características, como pode ser observado na Fig. 7. Os motores CC são alimentados por uma fonte de energia contínua, como o próprio nome sugere, que é geralmente uma bateria ou um rectificador. Fisicamente, os elétrons percorrem um único sentido neste tipo de alimentação. Já os motores CA são alimentados por uma fonte de corrente alternada, geralmente fornecida pela rede eléctrica de energia, onde o fluxo de elétrons alterna o sentido em função da mudança de polaridade do gerador da corrente. Por fim, os motores universais trabalham com as duas formas de corrente eléctrica.

Fonte: <https://traction.com/blog/motores-eletricos-entenda-a-funcionalidade-desse-ativo>



**Figura 7:** Fluxograma de especificações para motores eléctricos

Especificamente em relação aos motores AC, podemos subdividi-los em motores trifásicos e monofásicos. Motores trifásicos são alimentados por três fases de corrente eléctrica alternada, o que lhes permite operar de forma mais eficiente e suave em comparação com motores monofásicos, que são alimentados por apenas uma fase. Devido à sua eficiência, os motores trifásicos são amplamente utilizados em aplicações industriais de grande porte, como sistemas de produção de energia e bombas de grande capacidade. Por outro lado, os motores monofásicos são mais adequados para aplicações domésticas e pequenas, como ventiladores e ferramentas eléctricas, pois



são mais fáceis de instalar e requerem uma fonte de alimentação mais simples. Em resumo, a escolha entre os diferentes tipos de motores eléctricos, trifásicos ou monofásicos, depende da necessidade de potência, eficiência e facilidade de instalação para uma determinada aplicação. Uso de um inversor de frequência é importante em aplicações que requerem velocidades variadas, visto que ele permite controlar a velocidade e a eficiência do motor, garantindo o funcionamento otimizado em todos os momentos.

### ***2.3.1 Manutenção Preditiva***

Manutenção preditiva (também referida como preventiva baseada na condição) é aquela que indica as condições reais de funcionamento das máquinas com base em dados que indicam o seu nível de desgaste ou processo de degradação. Trata-se de um conjunto de processos para tentar estimar o tempo de vida útil dos componentes das máquinas e equipamentos e as condições para que esse tempo de vida seja bem aproveitado. Assim, atua-se com base na modificação de parâmetros de condição ou desempenho do equipamento, cujo acompanhamento obedece a uma sistemática. A manutenção preditiva pode ser comparada a uma inspeção sistemática para o acompanhamento das condições dos equipamentos.

A manutenção preventiva condicionada é realizada em função do estado dos componentes do equipamento, isto é, através de análises tribológicas (análises de óleos, análise da existência de detritos provenientes do desgaste das peças em contacto), termografia (processo que consiste na análise das temperaturas de diferentes órgãos ou componentes), ultra-sons (técnica que permite identificar diferenças de ruídos anormais em peças como rolamentos, etc.), análise de correntes eléctricas em motores de corrente alternada (CA), recolha e análise de vibrações (método que permite determinar o estado do componente facilitando assim a predição da sua avaria) (Dias, 2010). Para ser executada, a manutenção preditiva exige a utilização de aparelhos adequados, capazes de registar vários parâmetros e providenciar um histórico destes. Com base no conhecimento e análise dos fenómenos torna-se possível indicar, com antecedência, eventuais defeitos ou falhas nas máquinas e equipamentos. A manutenção preditiva, após a análise dos fenómenos, adota dois procedimentos para atacar os problemas detetados: estabelece um diagnóstico e efetua uma análise de tendências.

## CAPÍTULO III: METODOLOGIA

### *3.1 Máquinas Eléctricas Indústrias*

#### *3.1.1 Motores Eléctricos*

Este projeto foi idealizado para atender às necessidades reais de monitoramento de motores eléctricos industriais críticos em uma fábrica de cimento, especificamente o **motor do moinho de cimento** e de **Ventilador industrial**. Ambos são motores de grande porte, com potência superior a 200 kW, grau de proteção IP55 e ventilação IC411, operando continuamente em condições ambientais severas. Essas máquinas exigem sistemas de diagnóstico e manutenção preditiva, uma vez que falhas inesperadas resultam em perdas operacionais significativas.



*Figura 8: Uns do motor em estudo (Motor de Ventiladores)*

*Fonte: Autor, 2025*

No entanto, devido às limitações físicas, logísticas e de segurança associadas ao ambiente laboratorial, não foi possível utilizar motores industriais do tipo motor de Ventilador **Industrial** e **motor de Moinho de Cimento** durante a fase experimental. Assim, optou-se por desenvolver um protótipo funcional em escala reduzida, utilizando motores eléctricos de pequeno porte (figura 9), com características eléctricas e mecânicas compatíveis com os sensores e dispositivos electrónicos embarcados no sistema.



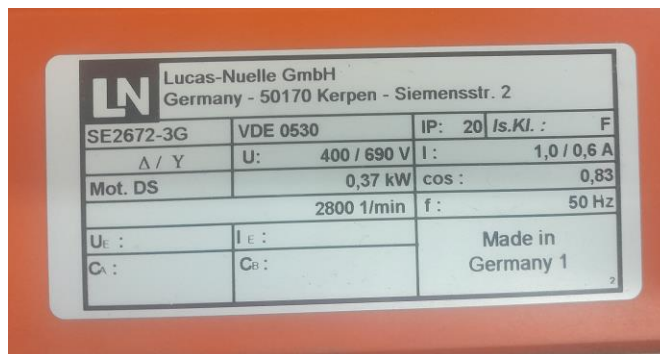
**Figura 9:** Motor trifásico para prototipagem

**Fonte:** Autor, 2025

O principal objetivo deste projeto é implementar uma gestão inteligente e eficiente dos motores elétricos críticos de uma fábrica de cimento, com foco na maximização da produtividade, redução de custos operacionais, prevenção de avarias e melhoria da manutenção preditiva. A monitorização contínua em tempo real permite detectar anomalias precocemente e agir rapidamente, evitando paragens não programadas. Foram seleccionados dois motores fundamentais para o processo: o motor de Ventiladores Indústrias e o do moinho de cimento. Estes motores desempenham papéis centrais na produção do clínquer e operam sob condições exigentes, sendo essenciais para o funcionamento contínuo da linha de produção. A falha de qualquer um deles representa grandes perdas de produção.

Para desenvolver o protótipo experimental do sistema baseado em IoT e gémeos digitais, foi utilizado no laboratório da Universidade Zambeze um motor de indução trifásico de baixa potência (0,37 kW, 400/690 V, 2800 RPM, IP20) figura 10, fabricado pela Lucas-Nuelle GmbH (Alemanha). Este motor foi integrado a um inversor de frequência e equipado com sensores de temperatura, vibração e ruído, permitindo simulações representativas do ambiente industrial. A recolha e análise de dados são realizadas em tempo real através de plataformas SCADA e outras

ferramentas de visualização, validando assim o conceito de gêmeo digital num ambiente controlado.



**Figura 10:** Chapa característica do motor da Lucas-Nuelle

**Fonte:** Autor, 2025

Para garantir um sistema de monitorização eficiente e fiável, é essencial proceder à selecção criteriosa dos sensores mais adequados, capazes de recolher dados relevantes sobre o desempenho dos motores eléctricos industriais. Entre os sensores mais importantes a se considerar encontram-se os sensores de temperatura, vibração, corrente eléctrica e velocidade de rotação. Estes dispositivos podem ser instalados directamente sobre os motores ou integrados nos seus respectivos sistemas de controlo, permitindo a obtenção de dados em tempo real sobre as condições de funcionamento. Para além da instrumentação, os próprios motores eléctricos possuem características técnicas específicas que variam conforme o modelo e a aplicação a que se destinam.

Para a fase experimental que engloba a criação de um protótipo, irá se criar um Sistema Supervisorio para facilitar o monitoramento dos parâmetros e obtenção dos resultados. De salientar que a proposta não exige a substituição do SCADA existente na fábrica em estudo, mas sim o seu reforço com novas funcionalidades e fontes de dados, através da ligação de sensores adicionais para medições de temperatura, vibração, corrente eléctrica e velocidade de rotação dos motores do forno rotativo e do moinho de cimento. A integração desses sensores com o sistema atual deve respeitar os protocolos industriais já suportados, como Modbus, OPC UA ou MQTT, garantindo a interoperabilidade e a continuidade operacional da instalação. A seguir, serão descritos os materiais físicos e virtuais utilizados, bem como os métodos adoptados para a implementação do Gémeo Digital, no âmbito do sistema de monitorização dos motores eléctricos críticos da Cimento.

A seleção dos materiais centrou-se principalmente nos componentes necessários para a aquisição e monitorização de dados operacionais, dada a natureza física do sistema e a necessidade de colocar os motores eléctricos em funcionamento real para permitir o desenvolvimento das restantes etapas do projecto. Assim, destaca-se a utilização de sensores de temperatura, vibração, bem como de dispositivos de aquisição de dados (DAQs), controladores lógicos programáveis (CLPs), módulos de comunicação e unidades de processamento. Paralelamente, foram também considerados os materiais virtuais, como software de simulação, plataformas SCADA, ferramentas de visualização 3D, bibliotecas de programação em Python e softwares de modelação utilizados para construir o Gémeo Digital.

No que respeita aos métodos de trabalho, será apresentado o planeamento do projecto, descrevendo-se os critérios que fundamentaram a escolha dos materiais, os objectivos de cada etapa e as metodologias aplicadas para integrar os componentes físicos e digitais. Serão ainda abordados problemas e desafios técnicos encontrados ao longo da implementação, como por exemplo limitações de compatibilidade, ruído nos sinais ou dificuldades de calibração de sensores, e as estratégias adoptadas para os ultrapassar. Adicionalmente, serão descritas as abordagens utilizadas para a coleta de dados em ambiente industrial, com destaque para a ligação ao sistema SCAD, garantindo-se a fidelidade e a representatividade dos dados recolhidos para alimentar e validar o Gémeo Digital.

### ***3.2 Inversor de frequência CFW-08***

O inversor de frequência CFW-08, da WEG, é um equipamento robusto e amplamente utilizado no mercado para o controlo de motores trifásicos com potências entre 0,25 e 20 cv. Neste projecto, o CFW-08 é aplicado para garantir o controlo preciso dos motores e a obtenção de dados eléctricos em tempo real, através do protocolo Modbus. Por defeito, este modelo não dispõe de comunicação Modbus integrada, sendo necessário acoplar o módulo opcional KRS-485 à sua parte frontal para permitir a ligação via RS-485. A comunicação ocorre por meio de telegramas Modbus RTU, que transportam bits e registos com as informações desejadas. Embora uma explicação detalhada sobre o protocolo esteja disponível no Apêndice A, o foco principal aqui recai sobre a configuração do CFW-08 e os métodos utilizados para aceder aos seus parâmetros e variáveis internas.



**Figura 11:** Inversor de frequência CFW-08

**Fonte:** [https://http2.mlstatic.com/D\\_NQ\\_NP\\_686323-MLA83334107614\\_042025-O.webp](https://http2.mlstatic.com/D_NQ_NP_686323-MLA83334107614_042025-O.webp)

O equipamento disponibiliza sete funções Modbus principais para essa finalidade, conforme descrito na Tabela 1 (WEG, 2009).

**Tabela 1:** Funções Modbus disponíveis no CFW-08

Função	Descrição	Código da Função	Tempo de Resposta
Read Coils	Leitura de bloco de bits internos ou bobinas	01	10 a 20ms
Read Holding Registers	Leitura de bloco de registradores do tipo holding.	03	10 a 20ms

<b>Write Single Coil</b>	Escrita em um único bit interno ou bobina.	05	10 a 20ms
<b>Write Single Register</b>	Escrita em um único registrador do tipo holding.	06	10 a 50ms
<b>Write Multiple Coils</b>	Escrita em bloco de bits internos ou bobinas.	15	10 a 20ms
<b>Write Multiple Registers</b>	Escrita em bloco de registradores do tipo holding.	16	10 a 50ms
<b>Read Device Identification</b>	Identificação do modelo do inversor.	43	10 a 20ms

**Fonte:** Autor, 2025

Dado que se trata de uma ligação a dois fios um para os dados e outro com os dados invertidos é necessária a utilização de um conversor para efetuar a leitura e interpretação das informações provenientes do inversor. Para esse fim, foi adquirido um conversor simples de USB para RS-485 (Fig. 3), permitindo estabelecer a comunicação entre o inversor CFW-08 e o computador.



**Figura 12:** Módulo Conversor USB para RS-485

**Fonte:** <https://www.amazon.com/Industrial-USB-RS485-Converter-Communication/dp/B081MB6PN2>

### **3.3 Controladores e Atuadores**

Apesar de o inversor de frequência neste caso, o modelo CFW-08 disponibilizar uma ampla gama de informações operacionais relativas ao funcionamento dos motores eléctricos, verificou-se que certos parâmetros essenciais para a modelação virtual detalhada não são diretamente fornecidos por este equipamento. Deste modo, a etapa subsequente do projeto consistiu na definição dos sensores, controladores e atuadores adicionais necessários para complementar a recolha de dados, tendo como finalidade a construção de um Gémeo Digital representativo e funcional do motor eléctrico. Para tal, foi realizado um levantamento técnico com base em normas europeias aplicáveis a máquinas eléctricas rotativas, nomeadamente a (ABNT, 2018) e IEC 60034 (Normas CEI aplicáveis à construção e ensaio de máquinas rotativas) e a Diretiva 2006/42/CE relativa às máquinas, bem como literatura técnica especializada, como o Guia de Especificação de Motores Eléctricos da WEG e Lucas-Nuelle GmbH (2025). Com base nesta análise, concluiu-se que três variáveis críticas — não disponibilizadas diretamente pelo inversor de frequência seriam imprescindíveis para o desenvolvimento preciso do modelo digital:

- Temperatura da carcaça do motor;
- Vibração em três eixos (X, Y, Z);
- Nível de ruído acústico.

Com estes requisitos em mente, foram definidos os materiais e dispositivos que serão utilizados para a aquisição precisa e em tempo real destes dados complementares. Estes incluirão sensores de vibração triaxial (acelerómetros), sensores de temperatura por infravermelhos ou termopares aplicados na carcaça, e microfones industriais com filtragem de ruído.

#### **3.3.1 Microcontrolador ESP32**

O microcontrolador ESP32, desenvolvido pela empresa Espressif Systems, com sede na China, foi seleccionado como o controlador central para o sistema de aquisição de dados dos motores eléctricos industriais. A escolha deste dispositivo foi fundamentada nas suas características técnicas altamente vantajosas, como a facilidade de programação, o baixo custo, o consumo energético reduzido, o formato compacto e, sobretudo, a sua capacidade de comunicação



sem fios integrada, através de Wi-Fi e Bluetooth, características indispensáveis para aplicações de Internet das Coisas (IoT) em ambientes industriais.



**Figura 13:** Microcontrolador ESP32

**Fonte:** [www.vidadesilicio.com.br/produto/esp32-esp-wroom-32-nodemcu](http://www.vidadesilicio.com.br/produto/esp32-esp-wroom-32-nodemcu)

Além disso, o facto de já existir familiaridade prévia com o dispositivo por parte da equipa de desenvolvimento, bem como a disponibilidade de unidades físicas para início imediato dos ensaios, reforçou a decisão da sua utilização neste projecto.

O ESP32 distingue-se como uma plataforma extremamente versátil, compatível com uma vasta gama de aplicações, desde automação industrial, robótica, e sistemas vestíveis (wearables), até aplicações domésticas inteligentes (domótica). O microcontrolador suporta diversas linguagens de programação, incluindo C/C++, Python, MicroPython e Lua, o que o torna acessível a diferentes perfis de programadores e investigadores. A programação do ESP32 foi realizada através da IDE (Ambiente de Desenvolvimento Integrado) do Arduino, uma plataforma open-source amplamente utilizada no desenvolvimento de protótipos electrónicos e aplicações IoT. Esta escolha permite uma integração facilitada com bibliotecas de sensores e comunicação, além de acelerar o processo

de prototipagem. A configuração detalhada da IDE, bibliotecas utilizadas e estrutura do código serão apresentadas na secção dedicada ao software do sistema.

Durante o processo de seleção do controlador mais adequado para a implementação do sistema de aquisição de dados dos motores industriais, foram considerados e avaliados diversos dispositivos, incluindo Controladores Lógicos Programáveis (CLPs) de diferentes fabricantes, bem como microcontroladores como o Raspberry Pi 3 e variantes do Arduino (UNO, Nano e Mega). Os CLPs, embora amplamente utilizados em ambientes industriais pela sua robustez e confiabilidade, foram descartados neste contexto específico devido à sua complexidade de programação e ao elevado custo associado. Dado que o sistema a ser desenvolvido visa apenas a monitorização de três parâmetros fundamentais (temperatura, vibração e ruído), e não requer uma grande quantidade de entradas e saídas digitais/analógicas, o uso de um CLP revelar-se-ia uma solução sobredimensionada, implicando num aumento desnecessário do custo total do projecto e na complexidade da sua implementação.

Relativamente ao Raspberry Pi 3, apesar das suas capacidades superiores de processamento e da possibilidade de executar sistemas operativos completos (como o Raspbian/Linux), a sua aplicação também foi considerada desadequada. Embora permita a integração directa de interfaces gráficas e lógica de programação local, a sua utilização implicaria custos mais elevados, bem como maiores requisitos em termos de energia, arrefecimento e manutenção. Além disso, partiu-se do pressuposto de que o ambiente industrial já dispõe de computadores de supervisão, tornando redundante a execução local do software de visualização no próprio dispositivo de aquisição de dados.

Quanto às variantes do Arduino, estas foram consideradas opções financeiramente vantajosas e adequadas para aplicações simples. No entanto, um dos factores determinantes na escolha final do controlador foi a necessidade de comunicação sem fios. Enquanto o Arduino, por si só, não possui conectividade Wi-Fi ou Bluetooth integrada, o ESP32 apresenta estas funcionalidades nativamente, eliminando a necessidade de módulos adicionais (como o ESP8266 ou módulos HC-05/HC-06), o que simplifica o hardware, reduz custos e melhora a fiabilidade do sistema, minimizando falhas causadas por cablagens excessivas ou conexões instáveis em protoboards.

Assim, após uma análise comparativa de custo-benefício, desempenho e simplicidade de integração, optou-se pela utilização do ESP32, um microcontrolador moderno, eficiente e altamente versátil, que combina a robustez necessária para aplicações industriais com a flexibilidade das tecnologias de comunicação sem fios, tornando-se a solução mais equilibrada para o projeto em questão.

### ***3.3.2 Sensores de Temperatura, Ruído e Vibração***

Para a monitorização precisa das condições operacionais dos motores eléctricos na fábrica, foram seleccionados sensores específicos com o objectivo de recolher dados relevantes sobre a temperatura da carcaça, o nível de ruído emitido e as vibrações em múltiplos eixos. Estes parâmetros são fundamentais para a análise do desempenho do motor e a implementação eficaz do Gémeo Digital, permitindo antecipar falhas e otimizar a manutenção preditiva. A escolha destes sensores baseou-se em critérios técnicos como precisão, robustez, compatibilidade com o microcontrolador ESP32, e viabilidade económica. Os detalhes técnicos, configurações de hardware e integração com o sistema SCADA e o Gémeo Digital serão descritos nas secções subsequentes.

#### ***3.3.2.1 Sensor de Temperatura DS18B20***

A monitorização da temperatura externa da carcaça dos motores eléctricos é de importância crítica, sobretudo no contexto industrial, onde os motores estão sujeitos a regimes de operação intensivos e ambientes agressivos. Esta medição é essencial para garantir a fiabilidade e a longevidade do equipamento, uma vez que o sobreaquecimento pode comprometer o sistema de isolamento dos enrolamentos do motor, acelerando o seu envelhecimento e reduzindo significativamente a sua vida útil. Segundo a norma **IEC 60085**, os motores eléctricos devem ser classificados de acordo com a sua classe térmica de isolamento, que define a temperatura máxima que o material isolante pode suportar continuamente sem sofrer deterioração significativa. Esta classificação é fundamental para garantir que o motor opera dentro dos limites de segurança térmica e é normalmente indicada na placa de características do equipamento.

Para a medição da temperatura foi seleccionado o sensor digital DS18B20, amplamente utilizado em aplicações de monitorização térmica devido à sua elevada precisão, estabilidade e facilidade de integração com sistemas microcontrolados. Este sensor permite leituras de

temperatura com uma resolução programável de 9 a 12 bits, apresentando valores com uma margem de erro reduzida ( $\pm 0.5\text{ }^{\circ}\text{C}$  em faixas comuns de operação), o que o torna apropriado para ambientes industriais. A recolha contínua da temperatura da carcaça permite detectar padrões anormais de aquecimento, prever falhas térmicas e activar estratégias de manutenção preditiva, contribuindo significativamente para a fiabilidade operacional e para a redução dos custos de manutenção e paragens imprevistas. Com base nas especificações técnicas dos subitens 6.3 e 6.4 da norma **NBR 17094-1:2018 (ABNT)**, em consonância com a norma internacional **IEC 60034-1**, a faixa de temperatura ambiente recomendada para o funcionamento seguro de motores eléctricos industriais é de  $0\text{ }^{\circ}\text{C}$  a  $40\text{ }^{\circ}\text{C}$ . Esta faixa deve ser rigorosamente considerada em aplicações críticas, como nos motores de acionamento do forno rotativo e do moinho de cru em fábricas de cimento, onde a solicitação térmica é intensa e contínua. Estes motores geralmente são fabricados com classe de isolamento térmico F ou H, devido às condições severas de operação. Para a classe F, o aumento máximo de temperatura permitido nos enrolamentos é de  $105\text{ }^{\circ}\text{C}$ , com uma tolerância adicional de  $10\text{ }^{\circ}\text{C}$  para pontos quentes, totalizando uma temperatura máxima admissível de  $155\text{ }^{\circ}\text{C}$ . Já na classe H, esses limites sobem para  $125\text{ }^{\circ}\text{C}$  de aumento máximo e até  $180\text{ }^{\circ}\text{C}$  de temperatura total nos enrolamentos. Contudo, para garantir maior confiabilidade e prolongar a vida útil do motor, é prática comum utilizar um limite operacional de segurança abaixo do valor máximo teórico. Neste contexto, adota-se como referência de alarme a temperatura de  $140\text{ }^{\circ}\text{C}$  para motores classe F e  $160\text{ }^{\circ}\text{C}$  para motores classe H.

O sensor já seleccionado, o sensor digital DS18B20, cuja faixa de operação vai de  $-55\text{ }^{\circ}\text{C}$  a  $+125\text{ }^{\circ}\text{C}$ . Embora este sensor não alcance os  $155\text{ }^{\circ}\text{C}$  ou  $180\text{ }^{\circ}\text{C}$  máximos de forma directa, ele é eficaz para a detecção precoce de sobreaquecimentos, especialmente se o sistema for configurado para emitir alertas a partir de  $110\text{--}120\text{ }^{\circ}\text{C}$ , zona já indicativa de aproximação de uma condição crítica. O DS18B20 oferece resolução ajustável de 9 a 12 bits, com precisão típica de  $\pm 0,5\text{ }^{\circ}\text{C}$  entre  $-10\text{ }^{\circ}\text{C}$  e  $+85\text{ }^{\circ}\text{C}$ , sendo compatível com ambientes industriais agressivos quando adequadamente encapsulado.



*Figura 14: Sensor de Temperatura DS18B20*

*Fonte: [www.amazon.com/HiLetgo-DS18B20-Temperature](http://www.amazon.com/HiLetgo-DS18B20-Temperature)*

### **3.3.2.2 Módulo Sensor de Som KY-038**

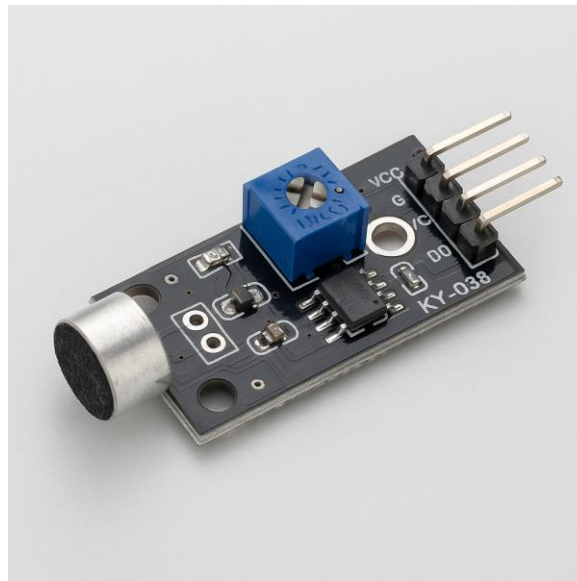
O controle de ruído em ambientes industriais é uma prática essencial não apenas para garantir a segurança e o conforto dos trabalhadores, mas também para detectar eventuais anormalidades no funcionamento de máquinas rotativas, como desequilíbrios, desalinhamentos, vibrações excessivas ou falhas incipientes nos rolamentos.

A norma europeia **IEC 60034-9** em coordenação com a **norma NBR7565(ABNT,1982)**, define os níveis máximos de pressão sonora (em dB(A)) emitidos por motores eléctricos rotativos em condições padronizadas. De acordo com a IEC 60034-9, os limites máximos de emissão sonora variam em função da potência nominal, velocidade de rotação, tipo de ventilação/resfriamento (por exemplo, IC01, IC411) e grau de proteção IP. Esses parâmetros são diretamente aplicáveis aos motores analisados, os quais operam com velocidades abaixo de 1800 RPM e potências típicas de até 6300 kW, dentro do escopo da norma. No motor de ventilador industrial, com uma potência nominal de 250 kW, velocidade de rotação de 990 RPM, grau de proteção IP55 (protegido contra

poeira e jatos d'água) e sistema de ventilação IC411, ou seja, auto ventilado com ventilador de resfriamento acoplado ao eixo. No caso do motor do moinho de cimento, submetido a esforços mecânicos intensos e funcionamento contínuo, com potência nominal de 400 kW, velocidade nominal de 990 RPM (motor de 6 polos), grau de proteção IP55 e sistema de ventilação IC411, que garante resfriamento eficiente por meio de ventilador externo acoplado, segundo a IEC 60034-9, motores com essas características possuem um limite superior de emissão sonora entre 86 dB(A) e 90 dB(A), medido a uma distância de 1 metro do invólucro do motor em condições de teste normalizadas. Exceder esses níveis pode indicar falhas típicas como desalinhamento do eixo, desequilíbrio do rotor, vibrações anormais, ou desgaste excessivo nos mancais e acoplamentos, todos fatores críticos em moinhos, dada a alta carga de impacto e operação sob condições severas. Considerando-se que o nível máximo estimado de ruído desses motores pode alcançar até 90 dB(A) em operação normal (conforme a norma IEC 60034-9), foi selecionado o sensor de som KY-038 como componente complementar de um sistema inteligente de vigilância sonora.

O KY-038 é um módulo compacto e de baixo custo, equipado com um microfone eletret, um comparador LM393, um potenciômetro (trimpot) de ajuste de sensibilidade e dois canais de saída (digital e analógico). Este sensor é capaz de detectar níveis sonoros na faixa aproximada de até 100 dB, com resposta eficaz em frequências próximas de 3 kHz, sendo sua faixa de resposta estendida até cerca de 6 kHz, o que o torna compatível com ruídos típicos gerados por vibrações e impactos metálicos em motores pesados. O funcionamento do sensor baseia-se na detecção de variações da pressão sonora no ambiente. Quando um som acima do limiar ajustado no trimpot é detectado, o sinal digital de saída muda de nível alto (sem som) para nível baixo (som detectado). Esse pulso digital pode ser interpretado por um microcontrolador como um evento de anomalia acústica, acionando alarmes, registros de log ou processos de shutdown de segurança. A saída analógica, por sua vez, fornece um sinal proporcional à intensidade do som captado, o que permite, com o uso de microcontroladores dotados de conversores analógico-digitais (ADCs) como o ESP32 ou Arduino, realizar uma leitura contínua e graduada da intensidade sonora. Essa leitura pode ser utilizada para mapear padrões de ruído ao longo do tempo, integrando-se a uma estratégia de manutenção preditiva com gêmeos digitais, onde dados acústicos são correlacionados com informações térmicas, elétricas e vibracionais dos motores em estudo.

Além disso, por ser altamente sensível a alterações no ambiente sonoro, o KY-038 pode ser estrategicamente posicionado próximo aos mancais ou acoplamentos dos motores para capturar ruídos de fricção, batidas secas ou ressonâncias, que geralmente antecedem falhas críticas. Assim, este sensor se insere como um componente simples, mas funcional, dentro de um ecossistema de monitoramento inteligente, contribuindo para a eficiência operacional, redução de tempo de parada não planejada e aumento da confiabilidade dos ativos industriais.



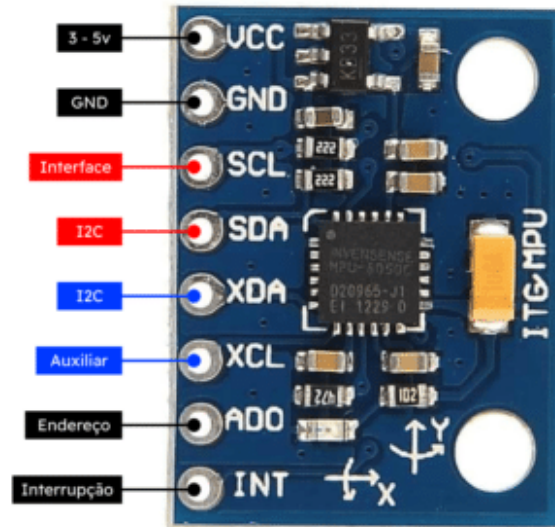
*Figura 15: Módulo Sensor de Som KY-038*

*Fonte:* [www.usinainfo.com.br/1021028-thickbox\\_default/sensor-de-som-ky-038-para-arduino.jpg](http://www.usinainfo.com.br/1021028-thickbox_default/sensor-de-som-ky-038-para-arduino.jpg)

### **3.3.2.3 Acelerômetro e Giroscópio MPU-6050 GY-521**

O monitoramento de vibrações é crucial para identificar falhas mecânicas em motores industriais de grande porte. Para isso, foi adotado o sensor MPU-6050 (figura 16) com o módulo GY-521, que combina acelerômetro e giroscópio triaxiais, permitindo a detecção de problemas como desbalanceamentos e falhas em rolamentos. Sua aplicação segue as normas **NBR 11390 (ABNT, 1990 e IEC 60034-14 (ABNT, 2011))**, que definem limites aceitáveis de vibração para motores CA e CC com potência até 50 MW. Com resolução de 16 bits e taxa de amostragem de até 1 kHz, o sensor oferece alta precisão e integração fácil com microcontroladores via I2C, como

ESP32 ou Arduino. Isso possibilita a análise de falhas, uso de algoritmos de aprendizado de máquina e visualização em dashboards, contribuindo para uma manutenção preditiva mais eficiente e maior confiabilidade dos equipamentos.



**Figura 16:** Acelerômetro e Giroscópio MPU-6050 GY-521

**Fonte:** <https://www.makerhero.com/wp-content/uploads/2017/07/Acelerometro-e-Giroscopio-3-Eixos-6-DOF-MPU-6050-9SS23-foto-prod-4.png>

### 3.4 Softwares

A implementação de um sistema de monitorização e controlo baseado em Internet das Coisas (IoT) exige a utilização de software adequado que actue como intermediário entre os componentes físicos do sistema nomeadamente os sensores, actuadores e controladores. Estes softwares são essenciais para garantir uma comunicação eficaz e fiável entre os dispositivos, permitindo uma coordenação precisa das acções, bem como a aquisição e o processamento contínuo dos dados. De forma geral, os softwares desempenham a função de interpretar os sinais de entrada captados pelos microcontroladores e convertê-los em comandos compreensíveis pelos actuadores, orientando assim o seu comportamento de acordo com as necessidades operacionais. Esta camada digital permite ainda a calibração, a configuração e a gestão dos dispositivos de campo, assegurando a sua interoperabilidade.



Com o objectivo de proporcionar uma visualização clara e em tempo real dos dados adquiridos tanto a partir do inversor de frequência como dos diversos sensores (temperatura, vibração, ruído, etc.) foi também desenvolvido um sistema SCADA (*Supervisory Control and Data Acquisition*). Este sistema supervísório facilita a monitorização centralizada, integrando funcionalidades cruciais como o registo de eventos, geração de alarmes, visualização gráfica de tendências e histórico de dados, contribuindo significativamente para a tomada de decisões operacionais e de manutenção. Para a programação dos microcontroladores utilizados no projecto, recorreu-se ao ambiente de desenvolvimento integrado (IDE) do Arduino, uma plataforma amplamente difundida e de fácil utilização, especialmente adequada para projectos académicos e prototipagem rápida. Para a construção do sistema supervísório SCADA, optou-se pelo ScadaBR, uma ferramenta de código aberto baseada em Java, que permite a criação de interfaces gráficas personalizáveis e a comunicação com equipamentos industriais através de protocolos como Modbus RTU/TCP, OPC e MQTT. As razões por detrás da escolha destas ferramentas, bem como detalhes adicionais sobre a sua configuração e integração, serão abordados nos tópicos seguintes.

### **3.4.1 *Arduíno IDE***

O Arduíno IDE (*Integrated Development Environment*) é uma plataforma de desenvolvimento de código aberto amplamente adoptada para a programação de placas da família Arduíno, bem como de outros microcontroladores compatíveis, como o ESP32. Trata-se de uma ferramenta essencial que simplifica significativamente o processo de criação, edição, compilação e transferência de código-fonte para os dispositivos físicos, tornando-a adequada tanto para principiantes como para utilizadores mais experientes. A interface do Arduíno IDE é intuitiva e orientada para a usabilidade, oferecendo funcionalidades robustas como realce de sintaxe, autocompletar, identificação de erros em tempo real e ferramentas integradas para a compilação e upload do código. Estes recursos aumentam a produtividade do programador e reduzem o tempo necessário para depuração e testes. A compilação do código é realizada com elevada fiabilidade, convertendo o script escrito em linguagens C/C++ para um formato binário que possa ser interpretado pelo microcontrolador. Após a compilação, o código pode ser facilmente transferido para a placa através de uma ligação via cabo USB, bastando ao utilizador seleccionar o modelo da placa e a porta de comunicação apropriada. (Arduíno 2025)

Entre as funcionalidades complementares mais destacadas, encontram-se:

- **Gestor de Placas:** permite instalar e configurar rapidamente os ficheiros específicos de diferentes modelos de hardware, incluindo controladores de terceiros;
- **Gestor de Bibliotecas:** oferece uma vasta gama de bibliotecas reutilizáveis que facilitam a integração de sensores, protocolos de comunicação (ex: Modbus), módulos de interface e outros periféricos. Também é possível importar bibliotecas externas manualmente;
- **Monitor Serial:** possibilita a leitura em tempo real de dados provenientes do microcontrolador, sendo extremamente útil para a monitorização de sensores, debug do sistema e análise de variáveis internas do código;
- **Compatibilidade com dispositivos não-Arduíno:** ao incluir o suporte ao ESP32 através da adição do repositório específico no gestor de placas, o IDE torna-se uma plataforma versátil para projectos de Internet das Coisas (IoT), automação industrial e sistemas embarcados. (ARDUINO, 2025).

O ESP32, por exemplo, pode ser facilmente programado no Arduíno IDE após a instalação da plataforma adequada. Diversas bibliotecas nativas e de terceiros já incluem suporte para este microcontrolador, facilitando a comunicação sem fios (Wi-Fi e Bluetooth), bem como a leitura de sensores e envio de dados para sistemas supervisórios como o SCADA. Devido à sua facilidade de uso, ampla documentação, grande comunidade de apoio e compatibilidade com os requisitos do presente projecto, o Arduíno IDE foi escolhido como a plataforma de desenvolvimento principal para a programação do sistema de aquisição de dados implementado neste trabalho.

### ***3.4.2 ScadaBR – Sistema de Controlo e Aquisição de Dados***

O ScadaBR é um software livre, gratuito e de código aberto, concebido para o desenvolvimento de sistemas de supervisão, controlo e aquisição de dados (SCADA), com forte aplicação em ambientes industriais, académicos e de investigação. Criado em 2006 por iniciativa da MCA Sistemas (actualmente Sensorweb), o ScadaBR foi desenvolvido com o apoio de diversas fundações e empresas do sector, com o objectivo de disponibilizar uma plataforma robusta e acessível, que fornecesse as mesmas funcionalidades dos sistemas SCADA proprietários, tradicionalmente dispendiosos. O ScadaBR foi desenhado para atender às necessidades de um

público diversificado, incluindo profissionais de automação, instituições de ensino técnico e superior, micro e pequenas empresas, entusiastas da electrónica e projectos de pesquisa, oferecendo uma solução eficaz para supervisão e controlo de máquinas, processos e sensores. Ao contrário de soluções SCADA comerciais, cujo custo pode ascender a dezenas de milhares de euros, o ScadaBR apresenta-se como uma alternativa economicamente viável, sem comprometer a funcionalidade. A sua natureza open-source constitui uma das principais vantagens competitivas: o utilizador pode adaptar o software às suas necessidades específicas, desenvolver novos módulos, integrar com sistemas já existentes e até contribuir com melhorias para a comunidade global do projecto (SCADABR, 2025).

Com uma base estimada de mais de 10 mil utilizadores activos, o sucesso do ScadaBR não se deve apenas à sua gratuidade, mas também à continuidade do seu desenvolvimento e ao investimento contínuo numa comunidade de utilizadores colaborativa. O seu site oficial disponibiliza documentação extensa em português, manuais de utilização, estudos de caso, cursos de formação e um fórum de discussão com mais de 6.500 membros e 4.000 publicações. Além da sua robustez e adaptabilidade, o ScadaBR apresenta uma interface web responsiva, compatível com computadores, tablets e smartphones, e é multiplataforma, podendo ser executado em Windows, Linux e macOS. Esta flexibilidade garante que os sistemas desenvolvidos com o ScadaBR possam ser mantidos e escalados independentemente da infraestrutura tecnológica da organização. Tendo em consideração todas estas características — nomeadamente a versatilidade na exportação e manipulação de dados, a compatibilidade com protocolos industriais, a facilidade de utilização e a disponibilidade de recursos técnicos em português — o ScadaBR foi seleccionado como ferramenta principal para a implementação do sistema supervisão deste trabalho.

## **CAPÍTULO IV: APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS**

O desenvolvimento deste trabalho alicerça-se na integração de três das quatro vertentes tecnológicas fundamentais dos Gémeos Digitais, conforme ilustrado na figura 2 de referência: Internet das Coisas (IoT), Inteligência Artificial (IA) e Computação em Nuvem. Estas tecnologias foram seleccionadas e exploradas com o intuito de construir um modelo digital representativo dos motores eléctricos monitorizados, alinhado com os princípios da Indústria 4.0. A primeira fase do desenvolvimento consistiu na implementação de uma infraestrutura de aquisição e transmissão de dados, com foco na comunicação bidireccional entre os sensores instalados nos motores, o microcontrolador ESP32 e o utilizador final. Esta camada é essencial para garantir a recolha contínua e fiável de variáveis críticas, como temperatura, vibração e ruído, permitindo o acompanhamento em tempo real do estado de operação dos motores.

Num segundo momento, foi iniciada a implementação de mecanismos de Inteligência Artificial, com o objectivo de conferir capacidades analíticas e preditivas ao sistema. A IA permite que os motores eléctricos se comportem como agentes inteligentes dentro do ecossistema industrial, aptos a partilhar activamente os seus dados, identificar padrões de comportamento e até antecipar falhas potenciais com base em modelos treinados com dados históricos. Complementarmente, a Computação em Nuvem é utilizada como repositório e plataforma de gestão dos dados recolhidos, permitindo o armazenamento seguro, escalável e remoto da informação. Dependendo da criticidade dos dados — por exemplo, valores de alarme ou eventos operacionais relevantes — estes podem ser armazenados localmente ou sincronizados com servidores na nuvem, assegurando disponibilidade e continuidade operacional. Esta arquitectura modular e escalável posiciona o protótipo como uma base sólida para futuras evoluções rumo a sistemas de manutenção preditiva industrial inteligentes, com base em gémeos digitais. Todos os passos realizados para o desenvolvimento do Gémeo Digital, desde a sua concepção inicial até à obtenção dos resultados experimentais, serão descritos de forma detalhada nas secções seguintes, com o objectivo de facilitar a compreensão do sistema desenvolvido, promover a sua análise técnica e possibilitar a sua replicação ou continuidade em projectos futuros.

Este capítulo está estruturado em três secções distintas, cada uma correspondente a um dos subsistemas fundamentais que compõem a arquitectura geral deste projecto:

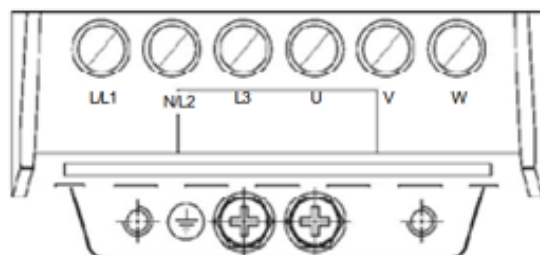
- **Sistema de Monitorização de Variáveis:** Nesta secção será abordada toda a concepção, configuração e implementação do sistema responsável pela recolha de dados provenientes dos sensores instalados nos motores eléctricos. Serão apresentados os componentes utilizados, os circuitos desenvolvidos, bem como os códigos-fonte programados no microcontrolador para garantir uma aquisição eficiente e fiável dos dados.
- **Sistema Supervisório:** Esta parte trata do desenvolvimento e configuração do sistema SCADA utilizado para a monitorização e análise das variáveis críticas do processo. Serão descritas as ferramentas de visualização adoptadas, os data sources configurados, a definição dos parâmetros de alarme, e a estrutura gráfica criada para apresentar as informações em tempo real de forma clara e intuitiva.
- **Sistema de Inteligência Artificial:** Por fim, será apresentada a camada de inteligência artificial desenvolvida em colaboração com o engenheiro Filipe. Esta secção explora a integração deste sistema com os demais subsistemas, detalha os algoritmos e plataformas utilizados, e discute os objectivos pretendidos no contexto da análise preditiva e da autonomia inteligente do modelo digital.

#### ***4.1 Sistema de Monitorização de Variáveis***

O sistema de monitorização de variáveis é constituído por um inversor de frequência CFW-08 e um microcontrolador ESP32, ao qual estão conectados os sensores DS18B20 (temperatura), KY-038 (ruído) e MPU-6050 (vibração e aceleração nos três eixos). Este conjunto é responsável pela recolha de diversos dados críticos relacionados ao estado operacional do motor e ao ambiente de funcionamento dos próprios dispositivos, incluindo corrente e tensão de saída, velocidade, temperatura do dissipador, parâmetros nominais do motor, temperatura da carcaça, nível de ruído e níveis de vibração mecânica, entre outros.

A primeira etapa do desenvolvimento centrou-se na compreensão do funcionamento do inversor de frequência, com base no manual técnico da (WEG 2009), e na sua correcta parametrização para garantir compatibilidade com as características eléctricas do motor monitorizado. Após a correcta configuração e activação do inversor, a atenção foi direccionada para a montagem do circuito de aquisição, envolvendo a ligação física dos sensores ao ESP32 e a respectiva programação, utilizando a plataforma Arduino IDE. Esta fase foi essencial para

assegurar uma comunicação fiável e contínua entre os sensores e o microcontrolador. Os resultados obtidos com esta estrutura serão apresentados nas secções seguintes, demonstrando o desempenho do sistema de aquisição no contexto do modelo de Gémeo Digital proposto.

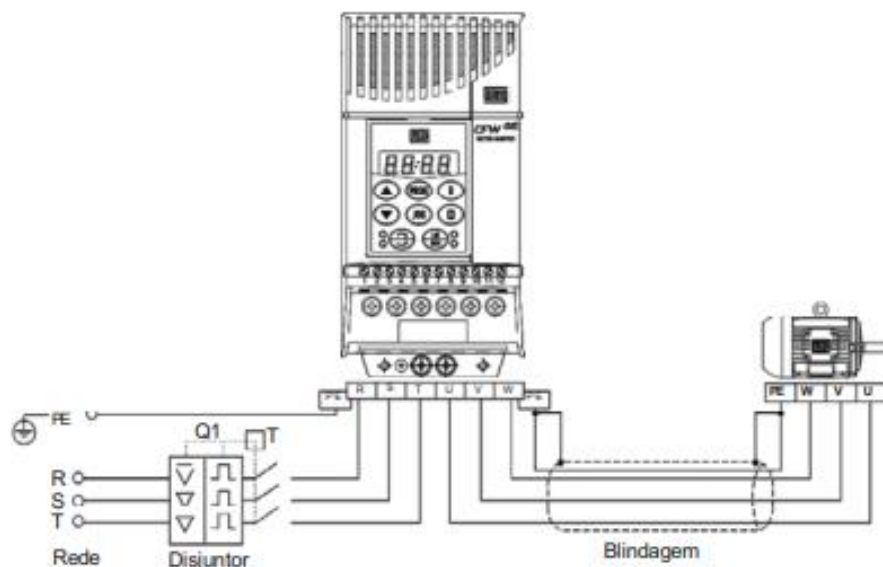


**Figura 17:** Bornes de potência do modelo utilizado do CFW-08

**Fonte:** <https://meloaut.com.br/wp-content/uploads/2021/02/Manual-do-Usuario-CFW08.pdf>

#### ***4.1.1 Alimentação e Ligação do Inversor de Frequência***

De acordo com as especificações técnicas constantes no manual do fabricante, os modelos da série CFW-08 com tensão nominal entre 200 e 240 V, à excepção de algumas versões específicas, podem funcionar em configuração bifásica (monofásica com neutro), sem comprometer a corrente nominal do equipamento. Nesses casos, a alimentação em corrente alternada (CA) pode ser aplicada a quaisquer dois dos três terminais de entrada do inversor, sendo ideal a utilização dos terminais identificados como L (fase) e N (neutro). Contudo, no presente projecto, o modelo adoptado do inversor de frequência é o CFW-08 2,6 A / 380–480 V, o qual requer alimentação trifásica de 380 V para o seu correto funcionamento. Dada essa exigência, torna-se imprescindível garantir uma ligação adequada aos três terminais de entrada de potência do equipamento, assegurando simultaneamente a correcta conexão ao sistema de terra (aterramento), de forma a proteger os dispositivos e os operadores contra falhas eléctricas. A figura 18 apresentada ilustra de forma clara a ligação dos bornes de potência e do sistema de aterramento necessário para o accionamento dos motores eléctricos através do inversor de frequência, conforme os requisitos técnicos definidos para a instalação na unidade fabril da empresa Cimentos.



**Figura 18:** Conexões de potência e aterramento

**Fonte:** <https://meloaut.com.br/wp-content/uploads/2021/02/Manual-do-Usuario-CFW08.pdf>

#### 4.1.2 Parametrização do Inversor CFW-08

Para que o inversor CFW-08 funcione correctamente com o motor conectado e permita comunicação via Modbus RTU, é essencial proceder à sua parametrização de acordo com as especificações técnicas. Inicialmente, os parâmetros foram configurados manualmente através da IHM frontal do inversor, conforme as instruções do manual. O primeiro passo consiste em definir o parâmetro P000 = 5, para desbloquear o acesso aos restantes parâmetros. Em seguida, o inversor deve ser colocado no modo de controlo vetorial (parâmetro P202), o que optimiza o desempenho do motor em termos de torque e velocidade.

Para activar a comunicação Modbus, foram configurados os seguintes parâmetros:

- P308: define o endereço do dispositivo na rede (ID).
- P312: selecciona o protocolo de comunicação (neste caso, Modbus RTU).
- P313 e P314: relacionados à função Watchdog, que permite acções de segurança em caso de falha na comunicação, mas não foram ajustados neste projecto.

A rede foi configurada com 9600 bps sem paridade, correspondente ao valor 1 em P312.

**Tabela 2:** *Parâmetros da Comunicação Serial*

Parâmetro	Função	Ajuste
P308	Endereço Serial	1
P312	Protocolo da Interface Serial	1
P313	Accão do Watchdog	-
P314	Tempo de Atuação do Watchdog	-

**Fonte:** Autor, 2025

Para garantir a segurança na operação dos motores, foram ajustados os parâmetros de aceleração e limitação de velocidade do inversor CFW-08. Os parâmetros P100 e P101 definem, respectivamente, o tempo de aceleração e desaceleração, determinando quanto tempo o motor leva para atingir a velocidade máxima ou parar. Já os parâmetros P133 e P134 estabelecem os limites mínimo e máximo da frequência de operação, o que reflecte directamente na velocidade do motor, visto que esta é proporcional à frequência aplicada. Estes limites asseguram que o motor não funcione abaixo de uma velocidade crítica, evitando falhas, nem acima do recomendado, prevenindo desgaste excessivo e prolongando a sua vida útil.

**Tabela 3:** *Parâmetros de Regulação*

Parâmetro	Função	Ajuste
<b>P100</b>	Tempo de Aceleração	5 s
<b>P101</b>	Tempo de Desaceleração	10 s
<b>P133</b>	Frequência Mínima ( $F_{min}$ )	3 Hz
<b>P134</b>	Frequência Máxima ( $F_{max}$ )	56 Hz

**Fonte:** Autor, 2025

#### **4.1.3 Conexão entre o ESP32 e os Sensores**

A interface entre o microcontrolador ESP32 e os sensores DS18B20 (temperatura), KY-038 (nível de ruído) e MPU-6050 (vibração e aceleração) requer uma integração correcta de hardware e software, assegurando que todos os dispositivos estejam devidamente conectados, alimentados e configurados para serem reconhecidos pelo sistema de aquisição de dados. O processo de



construção do circuito envolveu, numa fase inicial, a análise minuciosa das especificações técnicas dos pinos do ESP32 e dos sensores utilizados, com vista a garantir a correcta compatibilidade elétrica e lógica entre os componentes. Em seguida, procedeu-se à aquisição dos elementos adicionais necessários — como resistências pull-up, jumpers, protoboards ou PCBs — e à montagem do circuito, respeitando as recomendações de ligação fornecidas pelos respectivos fabricantes. A fiação entre os sensores e o microcontrolador foi realizada de forma organizada, assegurando ligações estáveis e minimizando interferências, especialmente no caso do sensor de vibração (MPU-6050), que é sensível a ruídos elétricos e mecânicos. Após a conclusão da montagem, os sensores foram posicionados estrategicamente na estrutura dos motores eléctricos, de modo a captar com precisão as variáveis desejadas. Idealmente, cada motor deve dispor de um circuito de monitorização independente, dado que, em ambientes industriais como o de uma fábrica de cimento, os motores encontram-se frequentemente dispersos em diferentes localizações. Essa separação física inviabiliza a utilização de um único microcontrolador para múltiplos motores, exigindo uma abordagem modular e descentralizada com comunicação remota. O ESP32, conforme já mencionado na secção anterior, é um microcontrolador da família **NodeMCU** que integra funcionalidades de Wi-Fi, Bluetooth clássico e Bluetooth **Low Energy (BLE)**, apresentando-se como uma solução altamente versátil e eficiente para aplicações em Internet das Coisas (IoT), automação e monitorização remota. Com uma arquitectura de dois núcleos **Xtensa®** de 32 bits, oferece elevado desempenho computacional, baixo consumo energético e uma excelente capacidade de comunicação sem fios. O módulo dispõe de um vasto conjunto de periféricos integrados, incluindo sensor de toque capacitivo, sensor de efeito Hall, interfaces para cartão SD, Ethernet, SPI de alta velocidade, **UART**, **I<sup>2</sup>S** e **I<sup>2</sup>C**, que o tornam extremamente adaptável às exigências de diferentes sistemas. Dispõe ainda de 25 pinos GPIO (**General Purpose Input/Output**), configuráveis como entradas e saídas digitais, canais analógicos com conversor ADC de 12 bits, saídas PWM ou canais de comunicação seriais, permitindo a interligação com sensores, actuadores e dispositivos industriais com grande flexibilidade. Esta versatilidade, aliada à sua robustez e fiabilidade, faz do ESP32 uma plataforma ideal para o desenvolvimento de sistemas inteligentes e conectados.

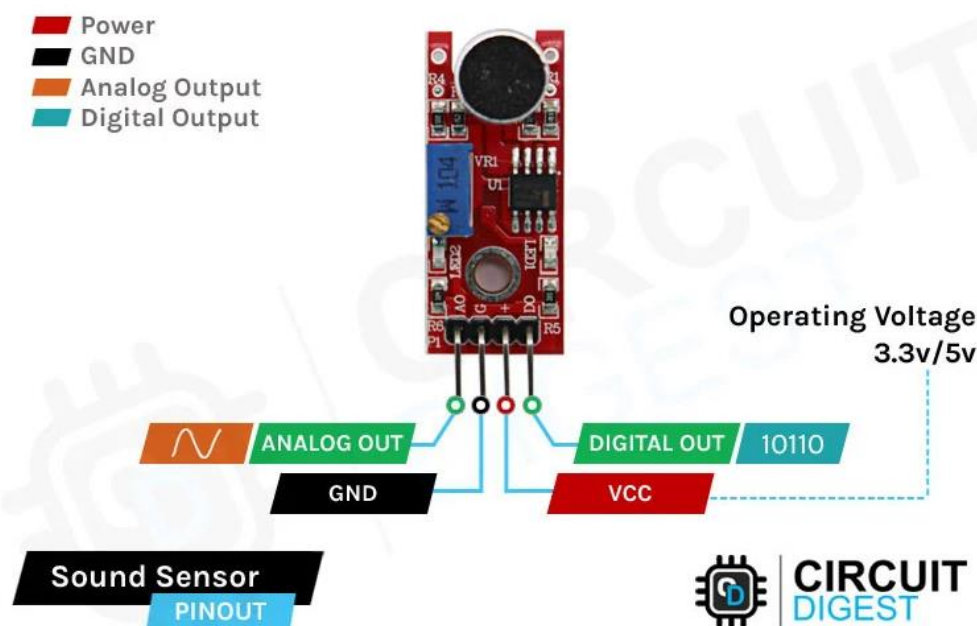




**Figura 20:** Pinagem do DS18B20

**Fonte:** <https://www.electronicwings.com/esp32/ds18b20-sensor-interfacing-with-esp32>

O KY-038 é um módulo eletrônico concebido para detetar e medir variações de som no ambiente, especialmente útil em aplicações que envolvem monitorização acústica. Este sensor dispõe de quatro terminais: **VCC** (alimentação), **GND** (terra), **A0** (saída analógica) e **D0** (saída digital), como ilustrado na figura. O terminal A0 permite a leitura contínua do nível de intensidade sonora por meio de sinais analógicos, enquanto o terminal D0 fornece uma saída digital que é ativada quando o som ultrapassa um limiar pré-estabelecido, definido por um potenciômetro incorporado no módulo. Para este projeto, como o objetivo principal é monitorizar as variações no ruído emitido pelo motor — e não apenas detectar picos sonoros — optou-se pela utilização da saída analógica A0, que foi ligada ao **pino GPIO 4** do ESP32. Esta configuração permite uma leitura mais precisa e contínua da intensidade sonora, essencial para análises detalhadas de funcionamento e diagnóstico de anomalias nos motores.



*Figura 21: Pinagem do sensor de som*

**Fonte:** <https://circuitdigest.com/microcontroller-projects/interface-ky038-sound-sensor-with-esp32>

Por fim, o MPU-6050 é um sensor versátil que integra, num único chip, um acelerômetro de três eixos e um giroscópio de três eixos, permitindo a medição simultânea de aceleração e rotação em tempo real. A comunicação deste sensor com o microcontrolador é realizada através do protocolo I2C (*Inter-Integrated Circuit*), um protocolo de comunicação serial que utiliza apenas duas linhas: SDA (Serial Data), responsável pela transmissão e recepção de dados, e SCL (*Serial Clock*), que fornece o sinal de relógio para sincronização entre os dispositivos. Além disso, o MPU-6050 inclui um barramento mestre I2C auxiliar, que permite a ligação a sensores adicionais, como magnetômetros, possibilitando a recolha de dados de múltiplos sensores de forma autónoma, sem intervenção direta do microcontrolador principal. No entanto, como o projeto em questão não exige a ligação de sensores I2C externos, apenas os pinos essenciais foram utilizados: **VCC** e **GND** para alimentação, e SCL e SDA para a comunicação de dados. A montagem do circuito iniciou-se com a soldadura da barra de pinos na placa do circuito integrado do sensor, garantindo ligações firmes e seguras. De seguida, procedeu-se à ligação dos pinos SCL e SDA aos **GPIOs 22** e **21** do **ESP32**, respetivamente, que são os pinos predefinidos para comunicação I2C neste

microcontrolador. Esta configuração assegura uma comunicação estável e eficaz entre o sensor e o sistema, fundamental para a monitorização de vibrações e movimentos nos motores industriais.

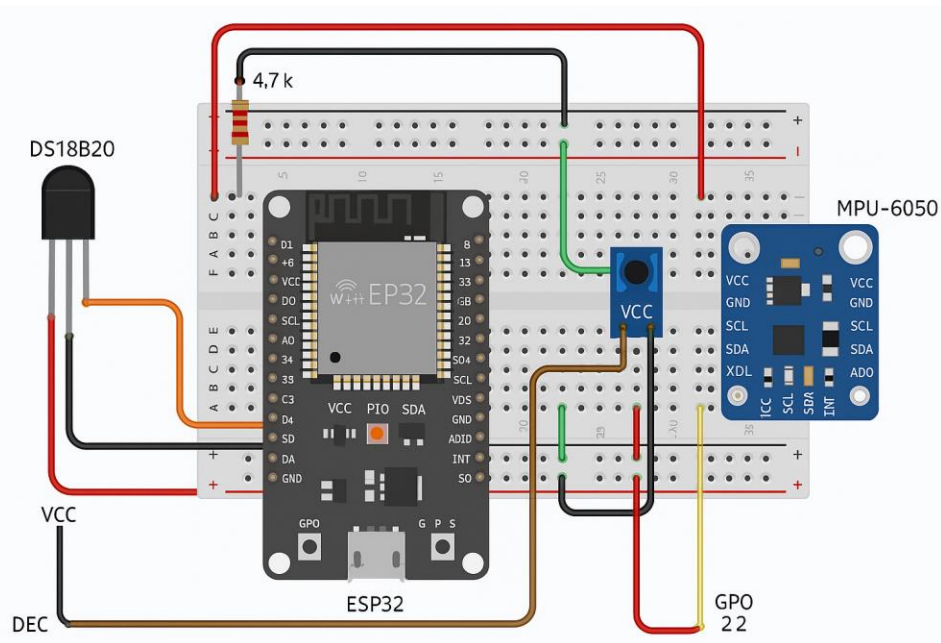
**Tabela 4:** Pinagem do acelerômetro MPU-6050

VCC	Fase (3,3V ou 5V)
GND	Neutro
SCL	Serial Clock pin para comunicação I2C
SDA	Serial Data pin para comunicação I2C
XCA	Usado para fazer a interface de outros sensores I2C com o MPU-6050
XCL	Usado para fazer a interface de outros sensores I2C com o MPU-6050
AD0	Use este pino para alterar o endereço I2C
INT	Pino de interrupção Pode ser usado para indicar que novos dados de medição estão disponíveis

**Fonte:** <https://pcbportugal.com/Pinagem-do-MPU6050-Um-sistema-micro-electro-mecanico-para-rastreio-de-movimentos-6D.html>

O circuito de aquisição de dados integra o microcontrolador ESP32 com três sensores essenciais: o DS18B20 (temperatura), o KY-038 (ruído) e o MPU-6050 (vibração e rotação). O ESP32 funciona como unidade central de processamento e comunicação. O sensor DS18B20 liga-se ao **GPIO 5** com alimentação por VCC e utiliza um resistor pull-up de 4,7 k $\Omega$  para garantir uma comunicação estável via protocolo **OneWire**. O sensor KY-038, conectado ao GPIO 4 pela saída analógica (A0), permite medir continuamente o nível de ruído; o pino digital não é utilizado. Já o sensor MPU-6050 comunica via interface I2C, com os pinos SDA e SCL ligados aos **GPIOs 21 e 22**, respetivamente. A montagem requer alimentação adequada e soldagem dos pinos do chip à protoboard para garantir boa conexão. Esta configuração possibilita a recolha de dados relevantes

(temperatura, som e vibração), fundamentais para a construção do Gêmeo Digital. A modelação do circuito foi realizada no software Fritzing.



*Figura 22: Esquema de Circuito Electrónico com ESP32*

#### 4.1.4 Programação dos dispositivos no Arduino IDE

A programação desenvolvida para captar os dados provenientes dos sensores e estabelecer comunicação com o sistema supervisor foi concebida de forma clara e intuitiva. O código completo encontra-se disponível no Anexo A, sendo o propósito desta subsecção descrever, com maior profundidade, os trechos mais relevantes do mesmo. À semelhança de qualquer outro programa baseado na plataforma Arduino, a estrutura do código é composta por duas funções principais: **setup()** e **loop()**. A função **setup()** é executada uma única vez aquando da inicialização do sistema, sendo responsável pela configuração dos dispositivos, inicialização de bibliotecas e definição de parâmetros essenciais. Já a função **loop()** é executada de forma contínua e repetitiva, contendo a lógica principal para a recolha de dados, processamento e envio das informações aos restantes módulos do sistema.

Foram utilizadas diversas bibliotecas fundamentais ao correcto funcionamento do programa, nomeadamente:

- **modbus-esp8266** – Responsável por permitir a comunicação via protocolo Modbus, tanto sobre redes Ethernet/Wi-Fi (Modbus TCP) como através de interfaces seriais (Modbus RTU). Esta biblioteca permite que a placa com ESP32 actue como mestre, escravo ou ambos.
- **Adafruit\_Sensor e Adafruit\_MPU6050** – A primeira define uma interface unificada para sensores, facilitando a interoperabilidade e consistência dos dados. A segunda é específica para o sensor MPU-6050, permitindo a leitura e interpretação dos dados do acelerómetro e giroscópio, bem como a comunicação via I2C.
- **DS18B20** – Biblioteca dedicada à leitura de sensores de temperatura digitais DS18B20. Suporta tanto a identificação automática de múltiplos sensores ligados na mesma linha quanto o endereçamento manual. Depende ainda da biblioteca *OneWire*, também necessária para o correcto funcionamento do barramento de dados de um fio (1-Wire).

Todas estas bibliotecas podem ser instaladas directamente através do **Library Manager do Arduino IDE** garantindo assim a sua versão mais recente e compatível. O código apresentado na secção seguinte respeita a estrutura do código original, com pequenas optimizações como a remoção de comentários redundantes e a compactação de blocos de código repetitivos, mantendo, no entanto, a clareza e funcionalidade para fins de estudo e manutenção futura. Avançando para a análise do código propriamente dito, numa primeira fase foram definidas algumas macros essenciais para a parametrização do sistema. Entre elas, destacam-se as seguintes:

- **SLAVE\_ID**: identifica o endereço Modbus atribuído ao microcontrolador ESP32;
- **KY038\_PIN** e **DS18B20\_PIN**: representam os pinos de entrada digital a que estão ligados, respectivamente, o sensor de som KY-038 e o sensor de temperatura DS18B20.

No caso específico deste projecto, foi atribuída ao ESP32 a identificação 1 como slave no protocolo Modbus, sendo que os sensores KY-038 e DS18B20 foram conectados às portas GPIO 4 e 5, respectivamente. Para o sensor MPU-6050, foi instanciado um objecto da classe **Adafruit\_MPU6050**, conforme disponibilizado pela biblioteca da **Adafruit**. Esta instância facilita a comunicação e manipulação dos dados captados pelo acelerómetro e giroscópio

embutidos no sensor. Em relação ao DS18B20, a biblioteca utilizada permite criar uma instância do sensor a partir de uma função construtora que recebe como argumento o número do pino ao qual o dispositivo está fisicamente ligado. Esta abordagem simplifica a identificação do sensor no barramento 1-Wire, sendo particularmente útil para a gestão de múltiplos sensores na mesma linha de dados, caso aplicável.

```
// === Bibliotecas necessárias ===
#include <ModbusRTU.h>
#include <OneWire.h>
#include <DS18B20.h>
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>

// === Definições de pinos e ID ===
#define SLAVE_ID      1
#define KY038_PIN     4
#define DS18B20_PIN  5

// === Objetos globais ===
ModbusRTU mb;                // Objeto Modbus
Adafruit_MPU6050 mpu;        // Objeto do sensor MPU-6050
OneWire oneWire(DS18B20_PIN); // Barramento OneWire
DS18B20 tempSensor(&oneWire); // Objeto do sensor de temperatura DS18B20
```

### Função setup()

A função setup() é responsável por inicializar os sensores e a comunicação via protocolo Modbus, além de definir os parâmetros fundamentais para o correto funcionamento do sistema. Logo no início da função, o pino ao qual o sensor de som KY-038 está ligado é configurado como entrada analógica. Em seguida, estabelece-se a comunicação série com o computador, definindo uma taxa de transmissão de 115200 bps, com 8 bits de dados, 1 bit de stop e sem paridade (configuração padrão 8N1). Esta configuração permite uma troca de dados eficiente entre o microcontrolador ESP32 e o terminal de desenvolvimento, possibilitando a visualização e depuração em tempo real das leituras dos sensores.



```
pinMode(KY038_PIN, INPUT);  
Serial.begin(115200, SERIAL_8N1);
```

Em seguida, procedeu-se à inicialização do protocolo Modbus, configurando o ESP32 como um dispositivo escravo com o endereço 1. Foram então definidos e adicionados os registos de retenção (holding registers), que são utilizados para armazenar os valores analógicos provenientes dos sensores. Para estas operações, recorreu-se às funções fornecidas pela biblioteca **modbus-esp8266**.

A criação de cada registo foi realizada através da função `addHreg()`, onde:

- O primeiro argumento corresponde ao deslocamento (offset) do registo, que define a sua posição lógica.
- O segundo argumento define o valor inicial armazenado nesse registo.

No total, foram criados 15 registos do tipo holding, todos com valor inicial de 0, permitindo o armazenamento e posterior transmissão das variáveis medidas pelo sistema de aquisição de dados.

```
mb.begin(&Serial);           // Inicializa a comunicação Modbus RTU via UART  
mb.slave(SLAVE_ID);          // Define o ESP32 como escravo com o ID especificado  
mb.addHreg(offset, valor_inicial); // Adiciona um registrador holding com offset e valor inicial
```

Por fim, procedeu-se à inicialização do sensor MPU-6050 recorrendo às funções disponibilizadas pela biblioteca **Adafruit\_MPU6050**. Durante essa configuração, definiu-se a faixa de medição do acelerómetro para  $\pm 2G$  (aproximadamente de  $-19,62$  a  $19,62 \text{ m/s}^2$ ), enquanto a gama do giroscópio foi ajustada para  $\pm 250^\circ/\text{s}$ . Adicionalmente, estabeleceu-se a largura de banda do filtro passa-baixo em 21 Hz, assegurando uma filtragem eficaz dos ruídos de alta frequência e permitindo uma leitura mais estável e fiável dos dados de movimento.

```
// Inicialização do sensor MPU-6050
if (!mpu.begin()) {
    Serial.println("Erro ao inicializar o MPU-6050!");
    while (1); // Aguarda indefinidamente se o sensor falhar
}
mpu.setAccelerometerRange(MPU6050_RANGE_2_G); // Acelerômetro com faixa de ±2G
mpu.setGyroRange(MPU6050_RANGE_250_DEG);      // Giroscópio com faixa de ±250°/s
mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);   // Filtro com largura de banda de 21Hz
```

## Função loop()

A função `loop()` é responsável pela execução contínua do programa, realizando a leitura em tempo real dos três sensores (temperatura, ruído e vibração) e o consequente armazenamento dos seus valores nos registos da rede Modbus.

O processo inicia-se pela leitura da temperatura com o sensor DS18B20, utilizando-se a função `getTempC()` da respetiva biblioteca. Esta função devolve o valor da temperatura em graus Celsius, armazenando-o numa variável do tipo **float** de 32 bits chamada `temperatura`. Contudo, como o protocolo Modbus opera com registos de 16 bits, é necessário realizar uma conversão para permitir o correto armazenamento do valor. A solução adotada consiste em aceder ao conteúdo binário da variável `temperatura` e reinterpretá-la como uma variável do tipo `uint32_t` (inteiro sem sinal de 32 bits), designada por `valor_temp`. Posteriormente, este valor de 32 bits é fragmentado em dois inteiros de 16 bits, denominados **reg\_temp1** e **reg\_temp2**. Cada um destes valores é então atribuído a um registo Modbus distinto. Desta forma, é possível reconstruir, posteriormente, o valor original da temperatura (float de 4 bytes) a partir dos dois registos de 16 bits, garantindo compatibilidade com o protocolo e preservando a precisão dos dados.

```
float temperatura = temp.getTempC();
uint32_t valor_temp = *((uint32_t*)&temperatura);
uint16_t reg_temp1 = valor_temp >> 16;
uint16_t reg_temp2 = valor_temp & 0xFFFF;
```

O sensor de som KY-038, ligado ao ESP32, fornece valores analógicos entre 0 e 1023, resultado da conversão analógico-digital de 10 bits. Para medir o nível sonoro, realiza-se uma amostragem durante 50 milissegundos, utilizando a função `millis()` para controlo temporal. Durante esse período, são registados os valores mínimo e máximo do sinal lido com `analogRead()`. Calcula-se então a diferença entre esses extremos (valor de pico a pico), que é convertida numa estimativa do nível de ruído em decibéis (dB), considerando os limites do **ADC** e a sensibilidade do sensor. O valor final, armazenado na variável `ruído`, pode ser enviado ao sistema SCADA para monitorização e activação de alertas, caso ultrapasse os limites definidos.

```
unsigned long startMillis = millis();
unsigned int sinal_max = 0;
unsigned int sinal_min = 1024;
float sinal_do_som = 0;

while (millis() - startMillis < 50) {
    int valor_anl = analogRead(KY038_PIN);
    if (valor_anl < 1024) {
        if (valor_anl > sinal_max) {
            sinal_max = valor_anl;
        } else if (valor_anl < sinal_min) {
            sinal_min = valor_anl;
        }
    }
}

sinal_do_som = sinal_max - sinal_min;
int ruído = map(sinal_do_som, 0, 1023, 0, 90);
```

No que diz respeito à leitura dos dados provenientes do sensor de vibração e rotação (MPU-6050), é essencial obter eventos actualizados do sensor através das funções disponibilizadas pela biblioteca `Adafruit_Sensor`. Esta biblioteca segue uma arquitectura de driver unificado, que recorre essencialmente a dois tipos de objectos: **sensor\_t** e **sensors\_event\_t**. Embora o **objecto sensor\_t** contenha informações estáticas sobre o sensor (como fabricante, tipo e resolução), a sua utilização não é obrigatória neste contexto. O foco recai sobre o **objecto sensors\_event\_t**, que é fundamental para encapsular uma leitura concreta do sensor – designada por evento – representando dados recolhidos num instante específico no tempo.

Deste modo, no código implementado, foram instanciados três objectos do tipo **sensors\_event\_t**: **a**, **g** e **temp**, os quais são responsáveis, respectivamente, pela aquisição de dados do acelerómetro (vibração nos três eixos), do giroscópio (rotação angular) e da temperatura interna do sensor MPU-6050. A função **getEvent** realiza a leitura dos valores brutos dos sensores (acelerómetro, giroscópio e termómetro), converte-os para unidades SI com escalas adequadas e atribui os resultados às variáveis do objeto **sensor\_event\_t**. Os dados do acelerómetro (em m/s<sup>2</sup>) são armazenados em **a.acceleration.x**, **a.acceleration.y** e **a.acceleration.z**, correspondentes aos eixos X, Y e Z. Em repouso, o eixo Z tipicamente mede  $\approx 9.81$  m/s<sup>2</sup> (aceleração gravítica), enquanto X e Y tendem a zero, salvo em casos de inclinação ou vibração. O giroscópio (em rad/s) armazena seus valores em **g.gyro.x**, **g.gyro.y** e **g.gyro.z**, representando as rotações nos três eixos. A temperatura (em °C) é registrada em **temp.temperature**. Todos os sensores fornecem dados em **float** de 32 bits, que são convertidos em dois inteiros de 16 bits para transmissão eficiente. É importante destacar que o acelerómetro detecta tanto forças estáticas (como a gravidade) quanto dinâmicas (como vibrações), exigindo compensação da gravidade em aplicações industriais para isolamento de vibrações relevantes.

```
// Declaração das variáveis para armazenar os eventos dos sensores
sensors_event_t a, g, temp;

// Leitura dos valores atuais do sensor MPU-6050 (acelerómetro, giroscópio e temperatura)
mpu.getEvent(&a, &g, &temp);

// Extrai o valor do acelerómetro no eixo X (em float32)
float acelerometro_x = a.acceleration.x;

// Converte o float32 para uint32_t (reinterpretação de bits)
uint32_t valor_acel_x = *((uint32_t*) &acelerometro_x);

// Divide o uint32_t em dois uint16_t:
// - reg_acel_x1 contém os 16 bits mais significativos (MSB)
// - reg_acel_x2 contém os 16 bits menos significativos (LSB)
uint16_t reg_acel_x1 = valor_acel_x >> 16;      // Desloca 16 bits para direita
uint16_t reg_acel_x2 = valor_acel_x & 0xFFFF;   // Máscara para pegar últimos 16 bits
```

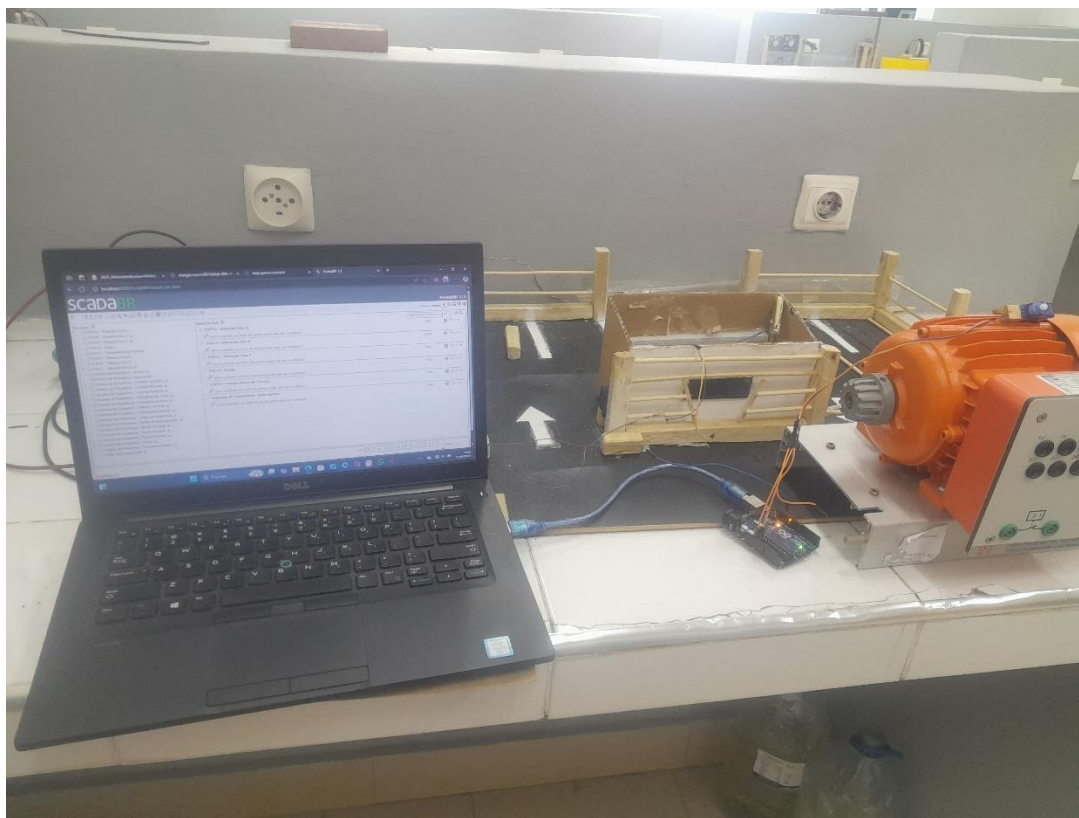
O sistema finaliza seu processo armazenando sequencialmente todos os valores processados nos registradores da rede Modbus. Para esta operação, utiliza-se a função **Hreg**, que segue a seguinte estrutura de parâmetros:

- Primeiro parâmetro: Offset (deslocamento) do registrador Modbus
- Segundo parâmetro: Variável de dados a ser armazenada

Este processo garante a correta organização dos dados na tabela de registradores do protocolo **Modbus**, permitindo sua posterior recuperação ou monitoramento por sistemas SCADA ou outros dispositivos mestre na rede.

```
void loop() {  
  mb.task();      // Processa requisições Modbus  
  mb.Hreg(0, sensorValue); // Atualiza registrador  
  yield();        // Mantém o sistema responsivo  
}
```

O sistema completo foi integrado com sucesso, conectando o ESP32, sensores e inversor de frequência, organizada com perfurações e cortes precisos para acomodar os componentes. Devido ao formato de conexão do ESP32, optou-se por uma instalação externa, utilizando alimentação eléctrica independente e cabos manga para as ligações aos sensores, garantindo flexibilidade e segurança. Todas as entradas e saídas de cabos foram concentradas em pontos estratégicos da caixa para manter a organização e facilitar a manutenção. Para futuras melhorias, recomenda-se o desenvolvimento de uma placa de interface personalizada para integrar o ESP32 internamente e a adoção de conectores industriais (como M12) para maior durabilidade nas conexões.



**Figura 23:** Sistema de Monitorização de Variáveis. **Fonte:** Autor, 2025

## **4.2 Sistema Supervisório**

O sistema supervisório foi desenvolvido com recurso à plataforma ScadaBR 1.2 (SCADABR, 2025), estando totalmente integrado com as variáveis fornecidas pelo sistema de aquisição de dados. A sua principal função consiste em permitir o monitorização contínua e controlo remoto dos parâmetros operacionais do inversor de frequência CFW-08, bem como dos valores obtidos pelos sensores DS18B20 (temperatura), MPU-6050 (vibração e rotação) e KY-038 (ruído sonoro).

Para a realização desta supervisão, foram configurados:

- Fontes de dados (Data Sources) do tipo Modbus RTU, responsáveis por estabelecer a comunicação entre o ScadaBR e o microcontrolador ESP32;
- Pontos de dados (Data Points), que representam cada uma das variáveis lidas em tempo real;

- Alarmes configuráveis, definidos com base em limites operacionais, de forma a alertar o utilizador quando qualquer parâmetro ultrapassar valores considerados críticos ou fora das condições normais de funcionamento;
- Interfaces gráficas amigáveis, construídas com painéis visuais que condensam as principais informações de forma clara e intuitiva num único ambiente, facilitando a leitura e interpretação pelo operador ou técnico de manutenção.

A seguir, será apresentada uma visão geral do sistema desenvolvido, bem como os detalhes técnicos e funcionais de cada um dos seus componentes, destacando a sua contribuição para o funcionamento e fiabilidade da aplicação.

#### ***4.2.1 Visão Geral do Sistema Supervisório***

O ScadaBR apresenta uma interface totalmente baseada na web, o que significa que, ao iniciar o software, é necessário aceder ao sistema através de um navegador de Internet à escolha. Após o carregamento, o utilizador é direccionado para uma página de autenticação, onde deverá introduzir um nome de utilizador e uma palavra-passe. Esta funcionalidade garante a gestão de utilizadores, permitindo a criação de múltiplos perfis com diferentes níveis de permissão, como por exemplo: operadores, técnicos, administradores ou clientes.

No âmbito deste projecto, e dado que a segurança de acessos não era o principal foco, optou-se por utilizar o utilizador padrão "**admin**", cuja palavra-passe é igual ao nome de utilizador. Este perfil possui acesso total a todas as funcionalidades e configurações do sistema desenvolvido. Após a autenticação, o sistema foi configurado para que a página inicial exibisse directamente a Watch List (Lista de Observação), em substituição da página padrão de ajuda. A **Watch List** é uma funcionalidade extremamente útil que apresenta, de forma automática e em tempo real, os valores actualizados dos pontos de dados (data points) aos quais o utilizador tem interesse em acompanhar. Esta funcionalidade elimina a necessidade de actualização manual da página (por exemplo, pressionando F5) e permite maior fluidez na monitorização contínua do sistema. É possível criar múltiplas listas de observação conforme a necessidade do utilizador, embora apenas uma possa ser visualizada de cada vez. No canto superior esquerdo da interface encontra-se uma

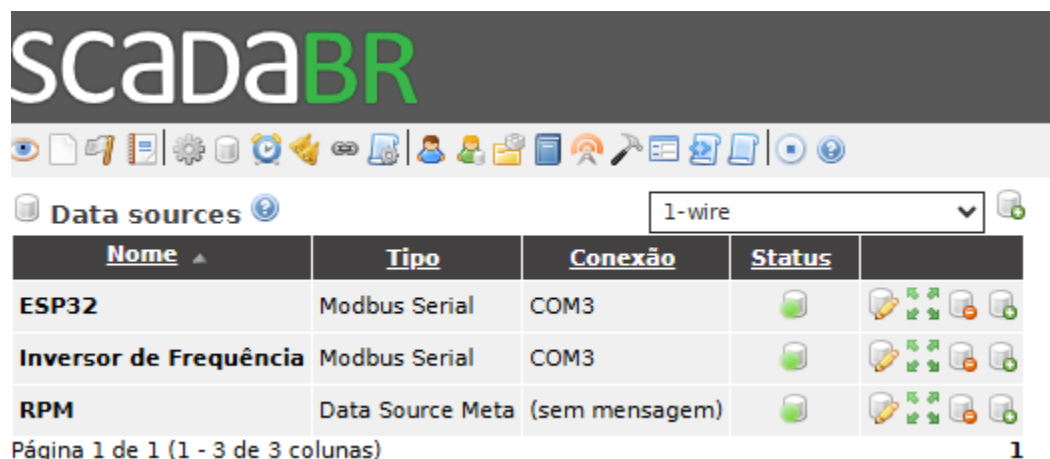
barra de navegação com ícones associados às principais funções do ScadaBR, junto ao logótipo da aplicação. Cada ícone dá acesso directo a funcionalidades específicas do sistema, como:

- **Fontes de dados (Data Sources)** — onde são configuradas as comunicações com os dispositivos físicos (via Modbus RTU, no caso deste projecto);
- **Representações gráficas** — onde se criam os painéis visuais com os indicadores das variáveis;
- **Alarmes** — configuráveis individualmente em cada ponto de dados, permitindo notificar o utilizador quando um parâmetro excede os valores operacionais aceitáveis.

Esta estrutura intuitiva permite um controlo completo e amigável sobre os dados recolhidos pelo sistema, contribuindo significativamente para a monitorização eficiente dos motores elétricos e seus sensores.

#### 4.2.2 Fontes de Dados (Data Sources)

A secção das fontes de dados representa um conjunto estruturado de elementos responsáveis por identificar e configurar todas as origens de informação dentro do sistema supervisorio. Cada fonte de dados possui atributos específicos, como o nome, o tipo, os parâmetros de ligação e o seu estado operacional (activo ou inactivo). As configurações de ligação variam consoante o tipo de dispositivo a ser monitorizado ou controlado, assegurando uma comunicação eficaz entre o ScadaBR e os respectivos equipamentos.



Nome	Tipo	Conexão	Status	
ESP32	Modbus Serial	COM3		
Inversor de Frequência	Modbus Serial	COM3		
RPM	Data Source Meta	(sem mensagem)		

Página 1 de 1 (1 - 3 de 3 columnas)

*Figura 24: Data sources do projeto. Fonte: Autor, 2025*



No âmbito deste projeto, foram definidos três data sources distintos, conforme ilustrado na Figura 24. Dois deles são do tipo Modbus Serial, criados para garantir a comunicação com o microcontrolador ESP32 (responsável pela leitura dos sensores) e com o inversor de frequência CFW-08. A terceira fonte é do tipo Meta Data Source, utilizada para o tratamento e transformação de variáveis, permitindo o cálculo e a exibição da velocidade do motor em rotações por minuto (RPM), com base nos dados obtidos do sistema.

O data source Modbus Serial é utilizado com o propósito de recolher dados provenientes de uma rede Modbus local, acessível através de uma interface de comunicação serial, seja RS232 ou RS485. Este tipo de fonte de dados baseia-se em consultas cíclicas, nas quais o mestre interroga os escravos em intervalos regulares para obter as variáveis de interesse. A sua configuração exige parâmetros como taxa de transmissão (baud rate), paridade, número de bits de dados, stop bits e o endereço do dispositivo escravo. Por outro lado, o data source do tipo Meta (ou “variável meta”) é assim designado devido à sua capacidade de manipular e combinar dados já existentes no sistema para gerar novos pontos de dados derivados. Em vez de recolher valores de dispositivos físicos, ele actua com base em expressões lógicas ou matemáticas aplicadas sobre outros data points. Esta funcionalidade é particularmente útil para cálculos adicionais, conversões de unidades, tratamento de dados complexos ou criação de indicadores compostos (como conversão de frequência para rotações por minuto - RPM).

## ESP32

Para compreender adequadamente o funcionamento do data source associado ao ESP32, é essencial analisar as suas propriedades e configurações específicas. Cada fonte de dados no ScadaBR requer um nome identificador, que pode ser atribuído livremente de acordo com a conveniência do utilizador. No caso particular do tipo Modbus Serial, o parâmetro Período de Actualização define o intervalo de tempo em que a rede Modbus será consultada para recolha de dados. Este valor influencia directamente a frequência com que as variáveis monitorizadas são actualizadas no sistema supervisão. Adicionalmente, os campos **Timeout** e **Retries** desempenham um papel importante na robustez da comunicação. O **timeout** estabelece o tempo máximo que o sistema aguardará por uma resposta do dispositivo Modbus após o envio de uma requisição. Se não houver resposta dentro desse intervalo, o sistema tentará reenviar a mesma

requisição um número de vezes definido pelo parâmetro retries. Estes mecanismos visam garantir a fiabilidade da comunicação, minimizando os efeitos de falhas momentâneas na rede.

A opção designada "Apenas quantidades contíguas" (Contiguous quantities only) define que a implementação do protocolo Modbus deverá restringir-se a realizar requisições apenas sobre blocos de registos consecutivos. Isto significa que, ao activar esta opção, o sistema evitará a combinação de múltiplos pedidos não contíguos numa única requisição, privilegiando uma comunicação mais simples e compatível com dispositivos cuja implementação Modbus é mais limitada. A comunicação via interface serial é regulada através de diversos parâmetros fundamentais, tais como: taxa de transmissão (baudrate), controlo de fluxo de entrada e saída, número de bits de dados (data bits), bits de paragem (stop bits) e configuração de paridade (parity). Estes parâmetros devem ser ajustados de forma coerente com a configuração do microcontrolador ESP32, que, por norma, é definida previamente na IDE do Arduino durante a programação. Destaca-se também a opção Echo, a qual pode ser activada em ligações RS-485 sempre que necessário. Esta funcionalidade reflecte os dados transmitidos de volta para o emissor, sendo útil em determinadas topologias de rede para verificação do envio.

No caso do data source criado para o ESP32, foram aplicadas as seguintes configurações:

- **Nome:** ESP32 (para facilitar a identificação da origem dos dados)
- **Período de actualização:** 5 segundos, garantindo uma monitorização quase em tempo real
- **Timeout:** também definido como 5 segundos, tendo em consideração possíveis atrasos na resposta do ESP32
- **Número de tentativas:** 2 repetições automáticas em caso de falha

Parâmetros de comunicação serial: conforme estabelecido no código-fonte carregado no microcontrolador via Arduino IDE.

**Propriedades do Modbus serial**

**Editar**

**Nome** Cópia de ESP32

**Export ID (XID)** DS\_712277

**Período de atualização** 5 segundo(s)

**Quantização** ☐

**Timeout (ms)** 5000

**Retentativas** 2

**Apenas quantidades contíguas** ☒

**Criar pontos de monitor de escravo** ☐

**Máxima contagem de leitura de bits** 2000

**Máxima contagem de leitura de registradores** 125

**Máxima contagem de escrita de registradores** 120

**Porta** COM3

**Baud rate** 115200

**Controle de fluxo de entrada** Nenhum

**Controle de fluxo de saída** Nenhum

**Data bits** 8

**Stop bits** 1

**Paridade** Nenhuma

**Codificação** RTU

**Echo** Desligado

**Simultaneidade** Função

**Figura 25:** Propriedades do data source ESP32. **Fonte:** Autor, 2025

Os data points, também denominados tags em alguns contextos industriais, correspondem às variáveis — numéricas ou alfanuméricas — utilizadas numa aplicação para monitorização e controlo de processos. Estas variáveis representam os dados adquiridos a partir de dispositivos físicos, e são configuradas com base em parâmetros específicos para garantir a leitura correcta dos valores. Tanto em redes seriais (como Modbus RTU) quanto em redes baseadas em IP (como Modbus TCP), os data points partilham atributos semelhantes para o mapeamento dos valores — como o tipo de dado, o offset (posição do registo), e o tipo de registo (Holding Register, Input Register, etc.).

No âmbito deste projecto, foram definidos oito data points associados ao data source "ESP32", representando as variáveis obtidas pelos sensores DS18B20 (temperatura), MPU-6050 (aceleração, rotação e temperatura interna) e KY-038 (nível de ruído). A figura correspondente (Fig. 26.) ilustra a estrutura criada. Todos os data points foram configurados como Holding Registers, e os offsets foram atribuídos em conformidade com a ordem estabelecida na programação do microcontrolador ESP32 (conforme descrito na secção). Importa referir que, com excepção do ponto de dados "Ruído", todas as outras variáveis são do tipo float (número real em 32 bits). Por essa razão, cada variável ocupa dois registos consecutivos de 16 bits, sendo necessário indicar no ScadaBR que o dado a ser interpretado combina o valor do offset seleccionado e o registo imediatamente seguinte. Este procedimento permite que o sistema reconstrua correctamente o valor float original. Já o data point "Ruído", por representar um valor inteiro de 16 bits sem sinal, utiliza apenas um único registo.

Data points					
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Rotação Eixo X	Numérico		1	Registrador holding	6
Rotação Eixo Y	Numérico		1	Registrador holding	8
Rotação Eixo Z	Numérico		1	Registrador holding	10
Ruído	Numérico		1	Registrador holding	12
Temperatura da Carcaça	Numérico		1	Registrador holding	13
Vibração Eixo X	Numérico		1	Registrador holding	0
Vibração Eixo Y	Numérico		1	Registrador holding	2
Vibração Eixo Z	Numérico		1	Registrador holding	4

**Figura 26:** Data points do ESP32. **Fonte:** Autor, 2025

## Inversor de Frequência (CFW-08)

Seguindo o mesmo procedimento usado para o ESP32, foi configurado um data source específico para o inversor de frequência CFW-08, identificado como "Inversor" no sistema supervisor. Este data source possui um período de actualização de 5 segundos e um tempo de espera (timeout) de 2,5 segundos, reflectindo a maior rapidez de resposta do equipamento em comparação com o ESP32. Não foram definidas tentativas de repetição, dada a fiabilidade da comunicação. Os parâmetros de comunicação serial foram ajustados de acordo com as

especificações do CFW-08. Activou-se ainda a opção "Apenas quantidades contíguas", para evitar erros de endereçamento no protocolo Modbus e garantir a leitura correcta de blocos de registradores sequenciais.

**Propriedades do Modbus serial**

Nome: Inversor de Frequência

Export ID (XID): DS\_503557

Período de atualização: 5 segundo(s)

Quantização: ☐

Timeout (ms): 2500

Retentativas: 2

Apenas quantidades contíguas: ☒

Criar pontos de monitor de escravo: ☐

Máxima contagem de leitura de bits: 2000

Máxima contagem de leitura de registradores: 125

Máxima contagem de escrita de registradores: 120

Porta: COM3

Baud rate: 9600

Controle de fluxo de entrada: Nenhum

Controle de fluxo de saída: Nenhum

Data bits: 8

Stop bits: 1

Paridade: Nenhuma

Codificação: RTU

Echo: Desligado

Simultaneidade: Função

Níveis de alarme de eventos

**Figura 27:** Propriedades inversor. *Fonte:* Autor, 2025

Foram configurados 16 data points no data source designado ao inversor de frequência CFW-08 (ver Fig. 28), os quais correspondem aos parâmetros descritos nas Tabelas 2 e 3 deste trabalho. Entre eles incluem-se: o parâmetro de acesso P000, os parâmetros de leitura P003, P005, P007 e P009, bem como o parâmetro de configuração P202, que define o tipo de controlo utilizado pelo inversor.

A definição e endereçamento destes pontos foram realizados em conformidade com as orientações especificadas no manual técnico do CFW-08 (WEG, 2009), respeitando a estrutura de comunicação Modbus do equipamento. Todos os data points foram configurados como valores inteiros sem sinal de 2 bytes (16 bits), visto que os parâmetros referidos são representados por palavras simples no protocolo Modbus RTU. Esta configuração permite uma leitura estável e eficiente dos valores operacionais do inversor, essenciais para o monitoramento em tempo real e eventual atuação em caso de desvios.

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Acesso	Numérico		1	Registrador holding	0
Auto-Ajuste	Numérico		1	Registrador holding	408
Corrente de Saída	Numérico		1	Registrador holding	3
Corrente-Nominal	Numérico		1	Registrador holding	401
Fator de Potência	Numérico		1	Registrador holding	407
Frequência de Saída	Numérico		1	Registrador holding	5
Frequência Nominal	Numérico		1	Registrador holding	403
Potencia Nominal	Numérico		1	Registrador holding	404
Rendimento Nominal	Numérico		1	Registrador holding	399
Tempo de Aceleração	Numérico		1	Registrador holding	100
Tempo de desaceleração	Numérico		1	Registrador holding	101
Tensão de Saída	Numérico		1	Registrador holding	7
Tensão Nominal	Numérico		1	Registrador holding	400
Tipo de Controle	Numérico		1	Registrador holding	202
Torque	Numérico		1	Registrador holding	9
Velocidade Nominal	Numérico		1	Registrador holding	402

**Figura 28:** Data points do Inversor. *Fonte:* Autor, 2025

## RPM

Por fim, o data source designado RPM não requer configurações adicionais de propriedades, uma vez que não está vinculado a uma comunicação externa directa, mas sim ao processamento interno de dados. Neste contexto, foi criado o data point denominado "Velocidade RPM", associado logicamente ao parâmetro P002 do inversor de frequência CFW-08. Contudo, devido à limitação de 16 bits imposta pelo protocolo de comunicação Modbus, o maior número inteiro que

pode ser representado num registrador é 65.536. Além disso, o CFW-08 utiliza a notação de ponto fixo, o que significa que os valores com casas decimais são transmitidos como inteiros multiplicados por 100 (por exemplo, uma corrente de 1,02 A é representada como 102 no registrador Modbus). Aplicando esse conceito ao parâmetro de RPM, o valor máximo real que pode ser representado diretamente seria 655,36 RPM — claramente insuficiente para as velocidades praticadas pelo motor industrial.

Para contornar essa limitação e obter valores realistas de rotação, recorreu-se a uma fonte de dados do tipo Meta no ScadaBR, que permite a realização de cálculos internos entre data points. Considerando as instruções do manual técnico da WEG (2009), identificou-se que o parâmetro P002 representa um valor proporcional à frequência (produto entre os parâmetros P208 e P005). Com base nisso, configurou-se a multiplicação entre esses dois parâmetros diretamente no ambiente do ScadaBR, obtendo-se assim o valor estimado da velocidade do motor em rotações por minuto (RPM). Esta lógica está ilustrada na Fig. 29 do presente trabalho.

The image shows two overlapping windows from the ScadaBR configuration software. The background window is titled 'Propriedades de data source Meta' and contains the following fields:

- Nome:** RPM
- Export ID (XID):** DS\_317517
- Níveis de alarme de eventos:**
  - data point de contexto desabilitado: Urgente
  - Erro de script: Urgente
  - Erro de tipo de resultado: Urgente
- Data points:** A table with columns 'Nome', 'Tipo de dado', and 'Status'. It contains one entry: 'Velocidade RPM' with 'Numérico' as the data type and a green status icon.

The foreground window is titled 'Detalhes do data point' and contains the following fields:

- Nome:** Velocidade RPM
- Export ID (XID):** DP\_253517
- Tipo de dado:** Numérico
- Configurável:** ☐
- Contexto do script:** ESP32 - Rotação Eixo X
- Script:** A text area containing the code:
 

```
x=p7.value*30;
return x;
```
- Atualizar evento:** Atualizar contexto
- Atraso de execução (segundos):** 0

Below the 'Contexto do script' dropdown, there is a table with three columns: 'Nome do ponto', 'Tipo de dado', and 'Var'.

Nome do ponto	Tipo de dado	Var
Inversor de Frequência - Frequência de Saída	Numérico	p7

**Figura 29:** Data Source e Data Point do RPM. *Fonte:* Autor, 2025

### Observação sobre o Parâmetro P208

O parâmetro P208 do inversor de frequência CFW-08 é utilizado exclusivamente quando o motor opera no modo de controlo vetorial. Em outras palavras, a medição e o cálculo da velocidade em rotações por minuto (RPM) apenas se justificam neste modo de funcionamento, não sendo necessária em configurações de controlo escalar (V/f). Importa ainda referir que a conversão de frequência (Hz) para velocidade rotacional (RPM) depende diretamente do número de pares de polos do motor elétrico. Esta relação é determinada pela fórmula:

$$\text{RPM} = \frac{120 \times f}{n} \text{ onde:}$$

- $f$  é a frequência elétrica em Hertz (Hz),
- $n$  é o número de polos do motor.

Dessa forma, para que a medição de RPM seja representativa da condição real do motor, é imprescindível que o valor correto de P208 (número de polos ou fator de escala correspondente) esteja devidamente parametrizado no inversor, conforme especificado pelo fabricante do motor.

#### 4.2.3 Sistema de Alarmes

No ScadaBR, o sistema de alarmes desempenha um papel fundamental na supervisão eficiente dos motores elétricos, permitindo a deteção e notificação imediata de anomalias operacionais. Existem dois tipos principais de alarmes: os alarmes do sistema, gerados automaticamente em casos como falhas de comunicação, endereços Modbus inválidos ou perda de ligação com dispositivos, e os alarmes definidos pelo utilizador, criados através de detectores de eventos configurados com base em condições específicas dos data points monitorizados. Estes alarmes podem acionar notificações visuais, registos históricos ou envio de alertas por e-mail/SMS, sendo possível configurar detectores simples ou compostos, que avaliam múltiplas variáveis simultaneamente ou com base em horários. Cada alarme possui um nível de severidade: Informativo, Aviso, Crítico ou Urgente consoante o grau de importância e a necessidade de intervenção. No âmbito deste projeto, foram definidos alarmes associados aos sensores DS18B20 (temperatura), KY-038 (nível de ruído) e MPU-6050 (vibração), com base em limites operacionais seguros para motores industriais. Os parâmetros considerados incluem: temperatura da carcaça



com limite superior de 75 °C; nível de ruído com limite máximo de 85 dB; vibração nos eixos X e Y com limites de  $\pm 2,5 \text{ m/s}^2$ ; e vibração no eixo Z com intervalo entre  $7,3 \text{ m/s}^2$  e  $12,3 \text{ m/s}^2$ . A configuração dos alarmes é realizada diretamente na interface web do ScadaBR, acedendo à Watch List, editando cada data point e personalizando os detectores de eventos com os respetivos limites, severidade e ações a executar quando as condições de alarme são verificadas.

#### ***4.2.4 Representações Gráficas***

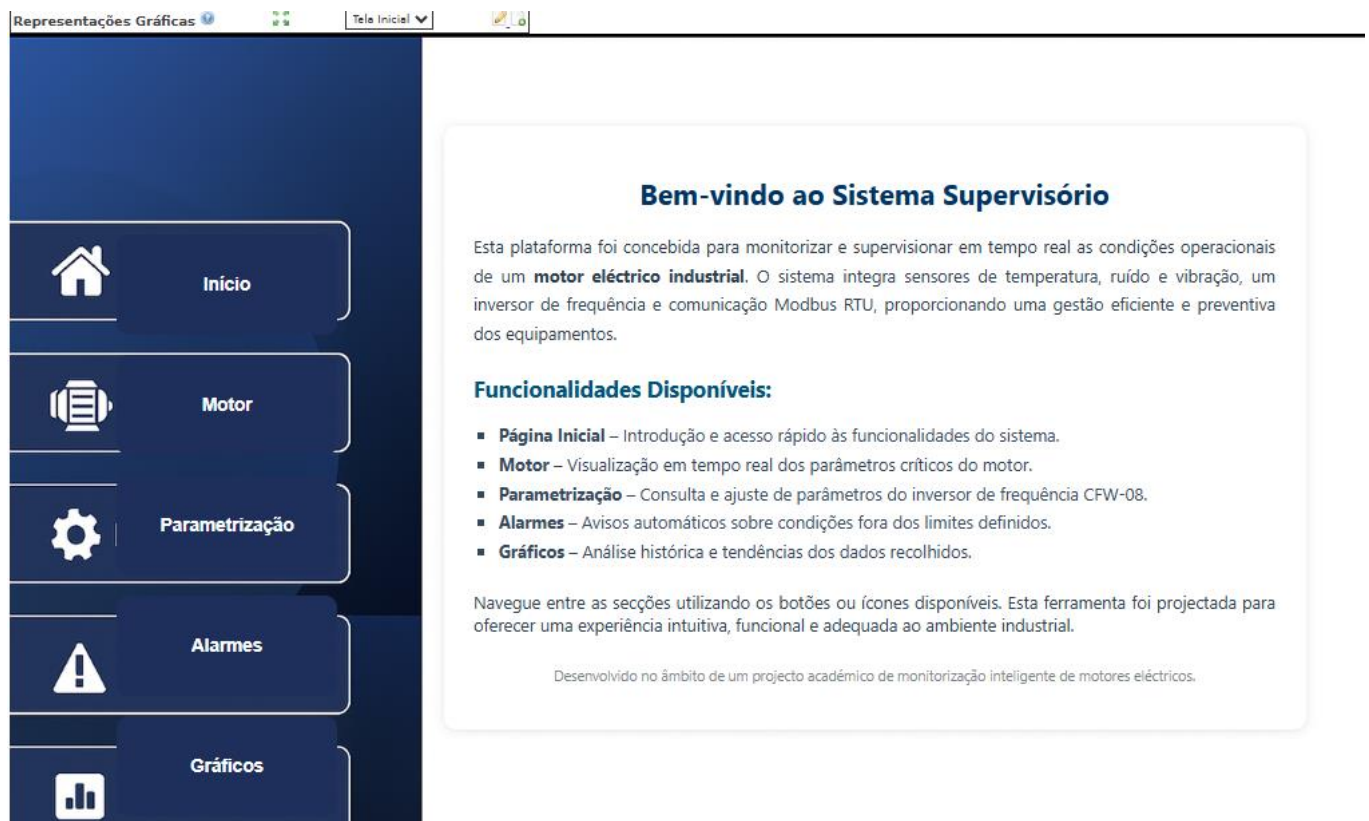
As representações gráficas no ScadaBR consistem em interfaces visuais que exibem os data points seleccionados, com actualizações automáticas dos seus valores e timestamps em tempo real. Cada utilizador pode criar múltiplas representações gráficas, embora apenas uma possa ser visualizada de cada vez. O ScadaBR disponibiliza uma ampla variedade de componentes gráficos destinados à visualização e supervisão de sistemas, incluindo gráficos em tempo real, tabelas, indicadores visuais, imagens com sobreposição de dados dinâmicos, botões e controlos interactivos, animações, bem como históricos com tendências. Esta flexibilidade permite a criação de interfaces personalizadas, adaptadas às necessidades específicas de cada projecto, proporcionando uma experiência de supervisão completa, intuitiva e eficiente.

A edição das representações gráficas é realizada através da página dedicada à configuração visual, onde é possível modificar todos os atributos da interface, bem como os pontos de dados associados. As visualizações podem conter componentes estáticos ou dinâmicos, dependendo da natureza dos dados envolvidos. Para adicionar conteúdo estático, é possível utilizar o componente do tipo HTML, que permite a inserção de trechos de código HTML válidos, incluindo JavaScript e a referência a classes de estilo definidas em folhas de estilo CSS. Já os componentes dinâmicos requerem a associação a um ou mais pontos de dados, sendo o Script para o Servidor um dos mais utilizados neste contexto. Este componente permite definir lógica de processamento personalizada através de código JavaScript, possibilitando maior controlo sobre o comportamento visual e funcional da aplicação.

#### **Pagina Inicial**

A página inicial, tem como principal objectivo introduzir o utilizador ao sistema supervisão, apresentando uma breve contextualização do projeto e das funcionalidades das demais telas. Trata-se de uma interface simples e intuitiva, concebida para facilitar a familiarização

progressiva do operador com o sistema. Do lado esquerdo, encontra-se o menu de navegação, que disponibiliza acesso às cinco telas desenvolvidas, destacando visualmente a página atualmente ativa. À direita, é exibido o conteúdo principal da tela, composto por instruções e descrições explicativas que orientam o utilizador quanto ao funcionamento geral do sistema.



**Figura 30:** Tela Inicial

**Fonte:** Autor, 2025

## Motor

A tela referente ao motor, ilustrada na Figura 31, tem como principal finalidade apresentar de forma gráfica, clara e objectiva os dados dos data points associados às propriedades do motor eléctrico monitoradas em tempo real. Esta interface oferece uma alternativa visual à **Watch list** “**Status do Motor**”, permitindo ao operador interpretar de forma mais intuitiva os parâmetros monitorizados. Mantendo a mesma estrutura da página inicial — com o menu de navegação

posicionado à esquerda e o conteúdo principal à direita —, a tela centraliza as informações em torno da imagem ilustrativa de um motor eléctrico, proporcionando uma experiência visual mais acessível e funcional ao utilizador.



*Figura 31: Tela do Motor*

*Fonte: Autor e Pinto Cunha, 2025*

## Parametrização

A tela de parametrização, ilustrada na Figura correspondente, tem como finalidade apresentar graficamente os parâmetros de configuração do inversor de frequência, de forma clara, objectiva e intuitiva. As informações exibidas correspondem à data **points** da **Watch list** “**Parametrização**”, permitindo a visualização e gestão dos valores associados à configuração do inversor modelo CFW-08. Tal como as demais telas, esta segue o mesmo esquema estrutural — com o menu de navegação à esquerda e o conteúdo ao centro e à direita — e integra uma imagem representativa do CFW-08, obtida a partir da página oficial do fabricante, com o intuito de reforçar

visualmente que os dados dizem respeito ao inversor. Importa destacar que os valores destes parâmetros podem ser modificados directamente através da interface, desde que o motor esteja desligado no momento da alteração, garantindo assim a segurança do equipamento e do operador.



**Figura 32:** Tela de Parametrização

**Fonte:** Autor e Pinto Cunha, 2025

## Alarmes

O seu principal objectivo é exibir os alarmes mais recentes gerados pelo sistema supervisor, proporcionando uma visão clara e imediata do estado de funcionamento dos equipamentos monitorizados. Nesta tela, é possível visualizar até 12 alarmes em simultâneo, os quais podem ser reconhecidos (acknowledged) ou silenciados conforme necessário. Esta funcionalidade é essencial para garantir a rápida identificação de ocorrências anómalas, facilitando

a acção correctiva por parte do operador e contribuindo para a segurança e eficiência do processo industrial.

🔔 Alarmes pendentes - 54 🔔 🔔

Reconhecer todos ✅ Silenciar todos 🔊

Id	Nível de alarme	Tempo	Mensagens	Tempo de inatividade	
184	🔔	08:51:55	Usuário admin entrou no sistema	Ativo 🔔	✅ 🔊
183	🔔	jun 18 17:51	Usuário admin entrou no sistema	08:45:50 - Retornou ao normal	✅ 🔊
182	🔔	jun 18 11:38	Usuário admin entrou no sistema	jun 18 16:22 - Retornou ao normal	✅ 🔊
181	🔔	jun 18 10:29	Usuário admin entrou no sistema	jun 18 11:04 - Retornou ao normal	✅ 🔊
180	🔔	jun 17 20:18	'Inversor de Frequência ': com.serotonin.modbus4j.exception.ModbusTransportException: com.serotonin.modbus4j.messaging.TimeoutException: request=SerialMessage [modbusMessage=ReadHoldingRegistersRequest [slaveId=1, getFunctionCode()=3, toString()=ReadNumericRequest [startOffset=0, numberOfRegisters=102]]]	Ativo 🔔	✅ 🔊
178	🔔	jun 17 18:49	'ESP32': Vibração Eixo X : com.serotonin.modbus4j.messaging.TimeoutException: request=SerialMessage [modbusMessage=com.serotonin.modbus4j.msg.WriteRegistersRequest [startOffset=0, numberOfRegisters=102]]	Ativo 🔔	✅ 🔊
177	🔔	jun 17 18:23	'ESP32': com.serotonin.modbus4j.exception.ModbusTransportException: com.serotonin.modbus4j.messaging.TimeoutException: request=SerialMessage [modbusMessage=ReadHoldingRegistersRequest [slaveId=1, getFunctionCode()=3, toString()=ReadNumericRequest [startOffset=0, numberOfRegisters=15]]]	Ativo 🔔	✅ 🔊
176	🔔	jun 17 18:20	'Inversor de Frequência ': Modbus Master: Erro de entrada/saída. Verifique sua conexão e porta de comunicação.	jun 18 10:17 - Retornou ao normal	✅ 🔊
175	🔔	jun 17 16:29	Usuário admin entrou no sistema	jun 17 20:05 - Retornou ao normal	✅ 🔊
174	🔔	jun 17 15:36	'ESP32': Modbus Master: Erro de entrada/saída. Verifique sua conexão e porta de comunicação.	jun 17 18:26 - Retornou ao normal	✅ 🔊
173	🔔	jun 17 10:01	Usuário admin entrou no sistema	jun 17 10:27 - Retornou ao normal	✅ 🔊
172	🔔	jun 17 09:33	'Inversor de Frequência ': com.serotonin.modbus4j.exception.ModbusTransportException: com.serotonin.modbus4j.messaging.TimeoutException: request=SerialMessage [modbusMessage=ReadHoldingRegistersRequest [slaveId=1, getFunctionCode()=3, toString()=ReadNumericRequest [startOffset=0, numberOfRegisters=102]]]	jun 17 15:36 - Retornou ao normal	✅ 🔊

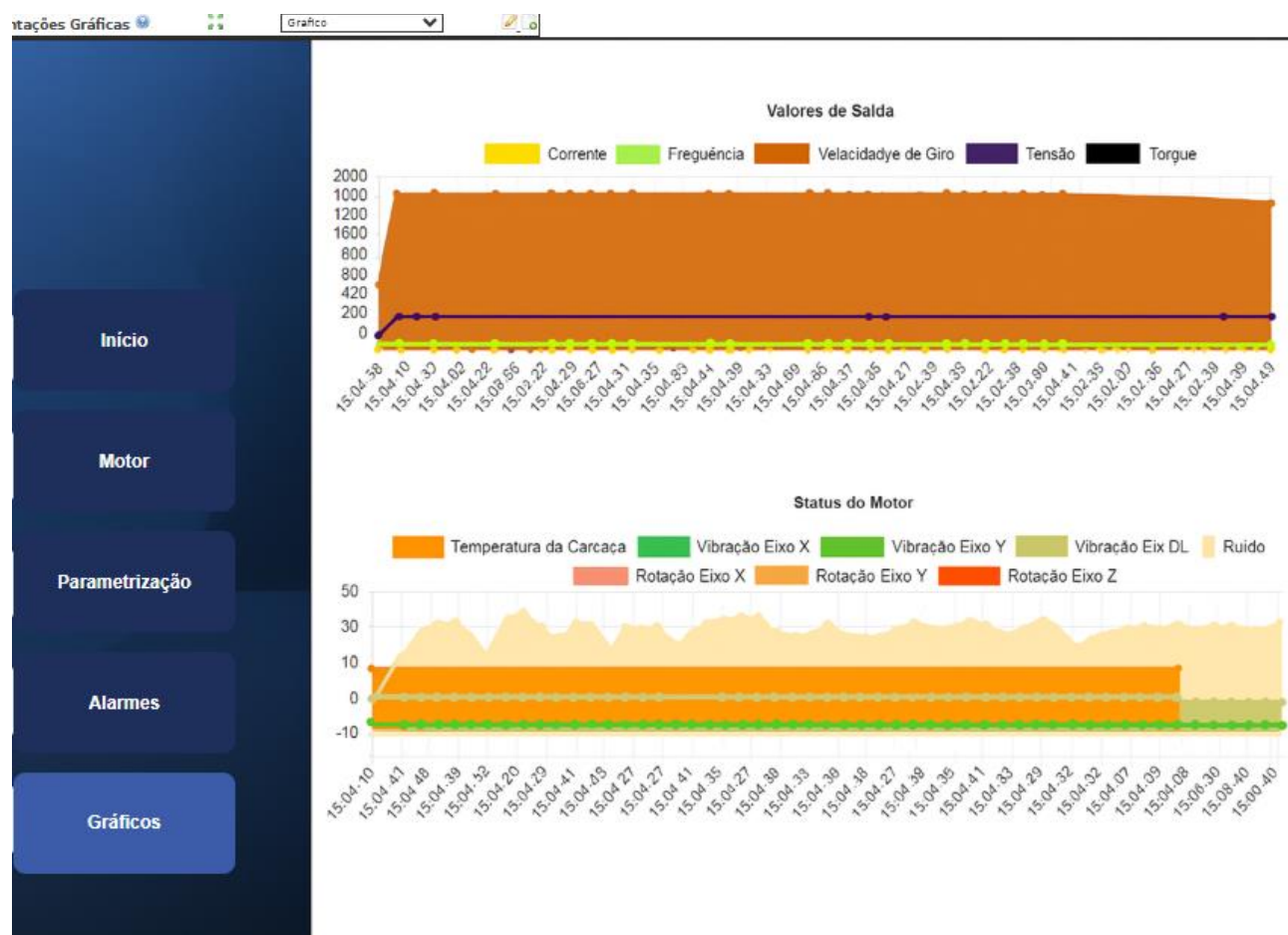
**Figura 33: Tela de Alarme**

*Fonte: Autor, 2025*

## Gráficos

A tela de gráficos, ilustrada com mais detalhe na Figura 34, segue o mesmo layout adoptado nas demais páginas do sistema, com navegação lateral e visualização centralizada. O seu propósito é oferecer uma análise visual da tendência das variáveis de status do motor ao longo do tempo, permitindo ao operador acompanhar o comportamento do sistema em tempo real. As informações foram organizadas em dois gráficos distintos, conforme a natureza dos dados — eléctrica e mecânica. O primeiro gráfico apresenta os dados eléctricos provenientes do inversor de frequência CFW-08, como velocidade, corrente e torque, todos essenciais para o controlo dinâmico do motor. O segundo gráfico mostra os dados adquiridos pelo ESP32, incluindo temperatura, ruído, vibração e rotação, representando os parâmetros de natureza mecânica e ambiental. Ambos os gráficos foram configurados para exibir uma janela de tempo de 5 minutos, no entanto, este intervalo pode ser ajustado nas definições conforme a necessidade do operador desde períodos mais curtos até

todo o histórico armazenado, o que é útil em auditorias e diagnósticos detalhados. Importa referir que as unidades das variáveis apresentadas não são necessariamente compatíveis entre si. Por exemplo, conforme se observa na Figura 34, os valores de velocidade de rotação podem atingir cerca de 1800 RPM, enquanto a corrente de saída permanece na ordem de 1 A. Para facilitar a leitura e análise, o sistema permite ocultar individualmente qualquer uma das variáveis através de um simples clique sobre a sua legenda colorida, possibilitando ao operador focar-se apenas nas informações de maior interesse, ou comparar selectivamente duas ou mais variáveis simultaneamente.



**Figura 34:** Tela de Gráficos

*Fonte:* Autor e Pinto Cunha, 2025

### ***4.3 Sistema de Inteligência na Indústria 4.0: Aplicação com Motores Eléctricos***

O trabalho de investigação, desenvolvido por Douglas Paula de Andrade (Brasil), tem como principal objectivo a incorporação de inteligência computacional na periferia (borda) da Indústria 4.0, utilizando motores eléctricos como caso de estudo. A proposta visa permitir a recolha, o armazenamento e a visualização de dados industriais em tempo real e com acesso ao histórico, o mais próximo possível da fonte de dados, ou seja, junto ao limite da rede. Ao descentralizar o processamento e posicionar parte da inteligência directamente na borda da rede, é possível reduzir o tráfego nas redes industriais, aumentar a eficiência, segurança e robustez dos sistemas e facilitar a detecção precoce de falhas, distribuindo o processo de monitorização ao longo da arquitectura industrial. A implementação deste sistema tem como finalidade otimizar o desempenho dos motores eléctricos e prolongar a sua vida útil, reduzindo significativamente os custos com manutenção e evitando paragens inesperadas no processo produtivo.

O projecto aborda quatro pilares principais:

- **Registo de Tags via Interface Web**

Foi desenvolvida uma interface web conectada a uma base de dados PostgreSQL. As tags dos dispositivos (variáveis monitoradas) não estão associadas directamente ao código-fonte, o que possibilita expansões e modificações nos equipamentos sem necessidade de interrupções ou alterações no código, promovendo flexibilidade e escalabilidade.

- **Controlador como Fonte de Dados**

Utiliza-se um dispositivo controlador como origem dos dados provenientes dos motores eléctricos. Este controlador estabelece a ponte entre o equipamento industrial e a interface inteligente do sistema.

- **Gateway Inteligente em Python**

O sistema inclui um gateway desenvolvido em Python com arquitectura orientada a objectos, que permite a integração de múltiplos protocolos de comunicação de forma modular. O gateway lê as informações armazenadas na base de dados e efectua a comunicação com os controladores, recorrendo aos endereços das tags previamente configuradas.

Este modelo de sistema inteligente demonstra como a aplicação de conceitos da Indústria 4.0, aliados à computação distribuída e à análise de dados em tempo real, pode transformar os motores eléctricos em unidades autónomas e inteligentes de produção, promovendo uma manutenção mais eficiente e uma operação mais fiável (ANDRADE, 2023).

### **Funcionamento do Sistema Inteligente: Integração de Dados, Armazenamento e Visualização**

De forma resumida, o funcionamento do sistema pode ser descrito da seguinte maneira: o código desenvolvido é responsável por ler as tags de interesse provenientes das fontes de dados, conforme as instruções armazenadas numa base de dados relacional. Os valores obtidos são posteriormente armazenados numa base de dados de séries temporais e disponibilizados para painéis de visualização e análise em tempo real.

Mais especificamente, a fonte de dados do sistema inteligente corresponde ao sistema de aquisição desenvolvido neste projecto. Os dispositivos ESP32 e o inversor de frequência CFW-08 actuam como elementos de campo, fornecendo informações cruciais sobre o funcionamento do motor eléctrico. Estes dados são então processados por um código em Python, que efectua a sua análise e interpreta o estado operacional do motor. O PostgreSQL, um sistema de gestão de bases de dados relacionais robusto e amplamente utilizado, é adoptado para registar as tags dos motores eléctricos e organizar a estrutura relacional do sistema de aquisição. Para interagir com essa base de dados, recorre-se à biblioteca psycopg, que permite efectuar leituras e gravações no PostgreSQL, incluindo no TimescaleDB — uma extensão do PostgreSQL optimizada para o armazenamento eficiente de séries temporais.

O TimescaleDB destaca-se por oferecer particionamento automático por data e chave, o que o torna particularmente adequado para aplicações industriais que geram grandes volumes de dados ao longo do tempo. Neste projecto, é necessário proceder à configuração do servidor TimescaleDB local, criar a base de dados e definir um utilizador com as respectivas credenciais, que serão utilizadas pelas ferramentas de visualização, como o Grafana e o Power BI.

**O Grafana** é uma plataforma de código aberto amplamente adoptada para a visualização interactiva de dados em tempo real. No contexto deste projecto, o Grafana é configurado para se conectar ao **TimescaleDB**, através da definição da URL, porta, nome da base de dados e



autenticação. A ferramenta converte os dados em gráficos dinâmicos e intuitivos, permitindo ao operador acompanhar, de forma visual, o desempenho do motor e os parâmetros monitorizados. Paralelamente, utiliza-se o Power BI como solução complementar, orientada para a gestão e análise estratégica dos dados. Com esta ferramenta, são criados dashboards personalizados, capazes de fornecer insights relevantes sobre os processos industriais. O Power BI também acede ao TimescaleDB, permitindo a organização, filtragem e visualização dos dados em relatórios técnicos e gerenciais, que podem apoiar decisões operacionais e de manutenção.

### **Integração de Bibliotecas Python e Inteligência na Monitorização de Motores**

Adicionalmente, o sistema desenvolvido recorre a bibliotecas especializadas da linguagem Python, nomeadamente o Flask e o Pymodbus. O Flask é um framework web minimalista, amplamente utilizado no desenvolvimento de aplicações leves e escaláveis. Neste contexto, foi utilizado para registar informações na base de dados PostgreSQL e permitir a visualização dos dados armazenados no TimescaleDB através de interfaces web simples e eficazes. Por sua vez, a biblioteca Pymodbus foi implementada para estabelecer comunicação com os dispositivos via protocolo Modbus, sendo fundamental para a leitura dos dados provenientes do sistema de aquisição. Esta biblioteca garante a integração fluida entre o código Python e os dispositivos industriais, tais como o ESP32 e o inversor de frequência CFW-08.

O código desenvolvido executa tarefas de monitorização inteligente das condições operacionais dos motores eléctricos, utilizando dados históricos para identificar falhas anteriores e avaliar a sua relevância. Esses dados são obtidos por meio de telemetria, através de sensores acoplados aos dispositivos, sendo posteriormente processados e estruturados para construção de um modelo digital do equipamento – base essencial para a criação de um gêmeo digital (*Digital Twin*). A monitorização contínua da telemetria permite avaliar o potencial de degradação do motor e calcular Indicadores-Chave de Desempenho (KPIs) relacionados à manutenção preditiva. Isto é alcançado pela combinação de dados históricos com informações de falhas, condições de operação e fatores de risco, proporcionando uma inteligência operacional contínua.

## CAPÍTULO V: CONCLUSÕES E SUGESTÕES

A proposta deste trabalho consistiu no desenvolvimento de um sistema para manutenção preditiva, otimização e monitorização de motores eléctricos com base no conceito de Gémeos Digitais. Através da implementação de três dos quatro pilares tecnológicos que sustentam este conceito Internet das Coisas (IoT), Inteligência Artificial (IA) e Computação em Nuvem, foi possível construir um modelo digital funcional e representativo dos motores analisados. O projeto integra um Sistema de Aquisição de Dados, um Sistema Supervisor e a um futuro não tao distante ira se incorporar um Sistema de Inteligência, este último baseado no trabalho de **Douglas Paula de Andrade**. Inicialmente, foram definidos os materiais e métodos de investigação, com foco na construção do sistema de aquisição de dados, dado o seu carácter físico e a necessidade de colocar os motores em funcionamento para permitir o desenvolvimento dos demais módulos. O inversor de frequência WEG CFW-08 foi empregado para o controlo preciso dos motores, possibilitando a obtenção de dados eléctricos em tempo real via protocolo Modbus, através de um conversor USB/RS-485. O microcontrolador ESP32 foi selecionado como unidade central, trabalhando em conjunto com os sensores DS18B20, KY-038 e MPU-6050, responsáveis pela medição da temperatura da carcaça, do nível de ruído e das vibrações nos três eixos, respetivamente. A programação dos dispositivos foi realizada na plataforma Arduino IDE, enquanto o sistema supervisor foi desenvolvido com recurso ao software ScadaBR, uma ferramenta de código aberto e de utilização gratuita.

Numa segunda fase, o trabalho centrou-se na implementação e integração dos sistemas de monitorização. O inversor de frequência foi devidamente alimentado, parametrizado e ligado aos motores, garantindo o seu controlo e comunicação eficaz via rede Modbus RTU. O ESP32 e os sensores associados foram corretamente programados, integrados e posicionados de acordo com as especificações dos pinos e a configuração eléctrica dos componentes. O código completo do sistema pode ser consultado no Anexo A. Paralelamente, o sistema supervisor foi estruturado com data sources para comunicação Modbus RTU, pontos de dados (data points) para armazenar variáveis, e alarmes configurados para detecção de anomalias. Foram também criadas interfaces gráficas intuitivas e funcionais, organizadas em cinco ecrãs principais, acessíveis através de um menu de navegação pensado para facilitar a interação com o utilizador.

Os resultados obtidos confirmam o êxito na concretização dos objetivos inicialmente propostos, evidenciando o potencial deste sistema para futuras melhorias e aplicações em contextos industriais mais amplos.

Na fase final do projeto, irá se integrar o sistema de aquisição de dados ao Sistema de Inteligência desenvolvido no trabalho de Douglas Paula de Andrade (Brasil), o qual, embora ainda em desenvolvimento, já apresenta uma estrutura funcional testada. Este sistema utiliza comunicação Modbus entre um código em Python, uma base de dados relacional e o sistema de aquisição, permitindo a leitura, armazenamento e visualização de variáveis em tempo real por meio de um banco de dados de séries temporais. Essa integração representa um passo significativo na concretização do conceito de Gêmeo Digital, demonstrando como as tecnologias aplicadas podem transformar os motores elétricos em sistemas inteligentes e conectados.

Apesar dos avanços, o trabalho reconhece que há espaço para melhorias. Propõe-se expandir a infraestrutura para permitir a monitorização simultânea de múltiplos motores, integrar os inversores diretamente aos ESP32 via pinos RX/TX, e unificar a comunicação através de uma única rede Modbus. Além disso, destaca-se a possibilidade de conexão remota utilizando Modbus TCP/IP ou LoRa, com o uso de módulos ESP32 LoRa TTGO, facilitando a comunicação entre motores e sistemas supervisórios. No que diz respeito à inteligência embarcada, o sistema ainda requer testes sob condições adversas, a fim de validar a sua capacidade de diagnóstico e reforçar a base de dados com ocorrências reais. Finalmente, para completar o conceito de Gêmeo Digital, recomenda-se a integração de modelação tridimensional através de softwares de realidade estendida, permitindo interação 3D com o modelo digital dos motores. Estas etapas futuras oferecem um campo promissor para investigação e inovação, contribuindo para o avanço tecnológico na indústria com soluções cada vez mais sofisticadas, eficazes e fiáveis.

## REFERÊNCIAS BIBLIOGRÁFICAS

ABNT. *IEC 60034-14 – Máquinas elétricas girantes – Parte 14: Medição, avaliação e limites de vibração mecânica de máquinas de altura de eixo igual ou superior a 56 mm*. 2011.

ABNT. *IEC 60085 – Isolação elétrica – Avaliação e designação térmicas*. Rio de Janeiro: ABNT, 2017.

ABNT. *NBR 11390 – Máquinas elétricas girantes – Medição, avaliação e limites da severidade de vibração mecânica de máquinas de altura de eixo igual ou superior a 56 mm*. 1990.

ABNT. *NBR 17094-1 – Máquinas elétricas girantes – Parte 1: Motores de indução trifásicos — requisitos*. Rio de Janeiro: ABNT, 2018.

ABNT. *NBR 7565 – Máquinas elétricas girantes – Limites de ruído*. Rio de Janeiro: ABNT, 1982.

ANDRADE, D. P. de. *Telemetry System in Electric Motors using Industry 4.0 Technologies*. In: **12th Brazilian Congress of Manufacturing Engineering**, 2023.

ARDUINO. *Tutorial do site oficial do software Arduino IDE*. 2023. Acesso em: 20 mai. 2023. disponível em: <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics#libraries>.

ATTARAN, M.; CELIK, B. G. *Digital Twin: Benefits, use cases, challenges, and opportunities*. *decision Analytics Journal*, v. 6, p. 100165, 2025. ISSN 2772-6622. DOI: <https://doi.org/10.1016/j.dajour.2025.100165>.

disponível em: <https://www.sciencedirect.com/science/article/pii/S277266222300005X>.

BALDISSARELLI, L.; FABRO, E. *Manutenção Preditiva na Indústria 4.0. Scientia cum Industria*, Universidade de Caxias do Sul, v. 7, n. 2, p. 12–22, 2019. ISSN 2318-5279.

CORREA, J. S.; RODRÍGUEZ CÓRDOBA, M. d. P.; LEYTON, J. D. *Efectos laborales vinculados al uso de técnicas de inteligencia artificial*. *Revista Universidad y Empresa*, v. 16, n. 26, p. 211–249, 2014. ISSN 0124-4639.

FERREIRA, B. A. P.; SERUFFO, M. C. D. R.; PIRES, Y. P. *Planejamento e construção de um protótipo de aplicativo mobile para visualização de dados de sistema de monitoramento de*

máquinas e equipamentos. *Revista Principia*, Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, v. 59, n. 3, p. 947–966, 2022. ISSN 1517-0306.

FRANCHI, C. *Inversores de Frequência: Teoria e Aplicações*. São Paulo: Saraiva Educação S.A., [s.d.]. ISBN 9788536511658. Disponível em: <https://books.google.com.br/books?id=QYqwDwAAQBAJ>.

GARCIA, E. *Introdução a Sistemas de Supervisão, Controle e Aquisição de Dados: SCADA*. Rio de Janeiro: Alta Books, 2019. ISBN 9788550809175. Disponível em: <https://books.google.com.br/books?id=vGocDwAAQBAJ>.

HALEEM, A.; JAVAID, M.; PRATAP SINGH, R.; SUMAN, R. Exploring the revolution in healthcare systems through the applications of digital twin technology. *Biomedical Technology*, v. 4, p. 28–38, 2023. ISSN 2949-723X. DOI: <https://doi.org/10.1016/j.bmt.2023.02.001>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2949723X23000065>.

JONES, D.; SNIDER, C.; NASSEHI, A.; YON, J.; HICKS, B. Characterising the Digital Twin: a systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, v. 29, p. 36–52, 2020. ISSN 1755-5817. DOI: <https://doi.org/10.1016/j.cirpj.2020.02.002>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1755581720300110>.

KUŠIĆ, K.; SCHUMANN, R.; IVANJKO, E. A digital twin in transportation: Real-time synergy of traffic data streams and simulation for virtualizing motorway dynamics. *Advanced engineering Informatics*, v. 55, p. 101858, 2023. ISSN 1474-0346. DOI: <https://doi.org/10.1016/j.aei.2025.101858>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1474034622003160>.

MARTINS, F. J.; FABRO, E. Uso do sensor inteligente na manutenção preditiva do motor de uma extrusora. *Scientia cum Industria*, v. 8, n. 2, p. 1–9, 2020. ISSN 2318-5279.

MARTINS, M. S.; PAULA, G. M. de; BOTELHO, M. de R. A. Inovações tecnológicas e Indústria 4.0 na siderurgia: difusão, estrutura de mercado e heterogeneidade intrassetorial. *Revista Brasileira de Inovação*, v. 20, e021006, 2021. ISSN 1677-2504. Citado na p. 14.

MARUTHI, G.; PANDURANGA VITTAL, K. Electrical Fault Detection in Three Phase Squirrel Cage Induction Motor by Vibration Analysis using MEMS Accelerometer. In: *2005 International*

*Conference on Power Electronics and Drives Systems*, 2005. v. 2, p. 838–843. DOI: 10.1109/PEDS.2005.1619804.

MODBUS. *The Modbus Organization*, 2014. Acesso em: 16 abr. 2025. Disponível em: <https://modbus.org/>.

MORAES, C. C.; CASTRUCCI, P. L. *Engenharia de Automação Industrial*. 2. ed. Rio de Janeiro: LTC, 2007. Citado nas pp.

NÓBREGA SOBRINHO, C. A.; SENA, A. P. C. de; LIMA FILHO, A. C. Estratégias para detecção de Falhas em Rolamentos de Motores de Indução Trifásicos. *Revista Principia*, Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, v. 1, n. 55, p. 108–118, 2021. ISSN 1517-0306.

OLIVEIRA, L. E. S. de; ÁLVARES, A. J. Desenvolvimento de um sistema para monitoramento e teleoperação de máquinas-ferramenta CNC via internet aderente à Indústria 4.0. *Revista Produção e Desenvolvimento*, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, v. 4, n. 1, p. 133–151, 2018. ISSN 2446-9580.

PANAROTTO, M.; ISAKSSON, O.; VIAL, V. Cost-efficient digital twins for design space exploration: A modular platform approach. *Computers in Industry*, v. 145, p. 103813, 2023. ISSN 0166-3615. DOI: <https://doi.org/10.1016/j.compind.2022.103813>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0166361522002093>.

WEG. *Manual do Inversor de Frequência CFW-08*. 2009. Acesso em: 29 abr. 2025. Disponível em: <https://meloaut.com.br/wp-content/uploads/2021/02/Manual-do-Usuario-CFW08.pdf>.

## APÊNDICES

## Apêndice A– Protocolo Modbus RTU

O protocolo Modbus inicialmente desenvolvido em 1979 pela fabricante de equipamentos Modicon, atualmente adquirida pela Schneider Electric, é um dos protocolos de comunicação mais antigos e até hoje mais utilizados em redes de CLPs para aquisição de sinais de instrumentos e comando de atuadores. Os direitos do protocolo foram transferidos para a Modbus Organization (MODBUS, 2014) em 2004, de forma que qualquer um pudesse utilizá-lo sem pagar royalties. Pelo fato de ser um protocolo aberto e por se adequar facilmente a diversos meios físicos, é um protocolo bastante famoso e amplamente difundido no ambiente industrial.

Em relação ao seu funcionamento, o protocolo transmite dados analógicos ou digitais por meio de redes seriais que conectam dispositivos eletrônicos, baseando-se em uma estrutura de mensagens compostas por bytes no modelo de comunicação do tipo mestre-escravo (ou servidor-cliente), onde pode haver até 247 escravos, mas somente um mestre, e nunca dois ou mais escravos podem ter o mesmo endereço. Estas mensagens são codificadas e existem dois modos de transmissão de dados especificados pelo protocolo que definem como os bits são transmitidos ao longo da rede e como os dados precisam ser empacotados e decodificados: o ASCII e o RTU. Os dois modos de transmissão não podem ser utilizados concomitantemente em uma mesma rede e, junto a eles, podem ser selecionados outros parâmetros da porta de comunicação serial, como taxa de transmissão, paridade, stop bits (MORAES;CASTRUCCI,2007).

O CFW-08 utiliza somente o modo RTU para comunicação, não possuindo, portanto, comunicação no modo ASCII. Devido a este fato, será focado e apresentado somente este modo de transmissão, onde o ASCII fica a título de curiosidade do leitor. No modo RTU, cada byte de dados é transmitido como sendo uma única palavra com seu valor diretamente. Em hexadecimal, onde cada palavra transmitida possui 1 start bit, oito bits de dados, 1 bit de paridade (opcional) e 1 stop bit (2 stop bits caso não se use bit de paridade) (WEG, 2009). Desta forma, a sequência de bits para transmissão de um byte é apresentada na **Fig. A.35**.

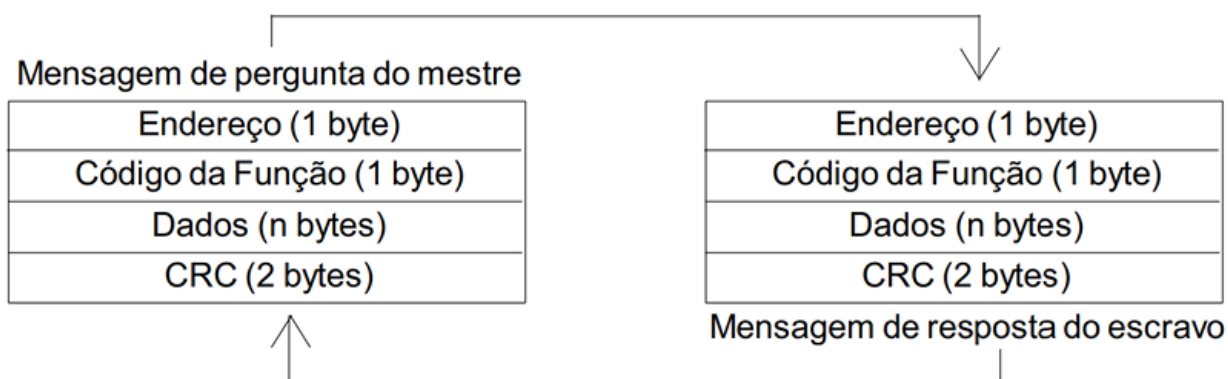


Start	B0	B1	B2	B3	B4	B5	B6	B7	Paridade ou Stop	Stop
-------	----	----	----	----	----	----	----	----	------------------	------

**Figura 35:** Transmissão de dados modo RTU

*Fonte:* (WEG, 2009)

Os telegramas são organizados utilizando uma mesma estrutura para a comunicação entre o mestre e o escravo em ambos os sentidos, sendo ela: Endereço, Código da Função, Dados e CRC. Apenas o campo de dados poderá ter tamanho variável, dependendo do que está sendo solicitado (WEG, 2009). Esta estrutura é ilustrada na **Fig. A.36**.



**Figura 36:** Estrutura das mensagens

*Fonte:* (WEG, 2009)

O campo do endereço contém a identificação do destinatário ou remetente da mensagem transmitida, visto que o endereço é sempre de um dispositivo escravo. O mestre envia um byte com o endereço do escravo que deve receber a mensagem ou pode enviar uma mensagem destinada ao endereço 0 (zero) para todos os escravos da rede. Já o escravo inicia o telegrama com o seu próprio endereço. O campo código da função especifica o tipo de serviço ou função solicitada ao escravo, no caso do CFW-08 apresentados na tabela 3.1. O Campo de dados contém a informação em si do telegrama, onde o formato e o conteúdo deste dependem da função utilizada e dos valores transmitidos. Por fim, o CRC é o campo para Checagem de erros de transmissão, formado por dois bytes, sendo que o primeiro transmitido é o byte menos significativo (CRC-), e em seguida o mais

significativo (CRC+). Não existe um carácter específico que indique o início ou o fim de um telegrama.

O tempo entre as mensagens é utilizado para identificar o começo e o término, onde a ausência de transmissão de dados na rede por um tempo mínimo de 3,5 vezes o tempo de transmissão de uma palavra de dados indica uma nova mensagem, sendo que após esse período a mensagem anterior chegou ao fim. A velocidade da comunicação pode ser ajustada a depender da aplicação, variando-se o Baud Rate da porta serial. O valor padrão utilizado quando não se tem um requisito ou especificação é 9600 bit/s, porém existem vários outros valores que podem ser utilizados. A tabela A.6 apresenta alguns valores comuns utilizados e disponíveis para configuração no CFW-08, relacionando-os com o tempo de transmissão de uma palavra de dados  $T_{11bits}$  e o intervalo mínimo de tempo entre uma mensagem e outra  $T_{3,5x}$ . É importante lembrar que todos dispositivos ligados a uma mesma rede devem estar com a mesma taxa de comunicação, visto que não havendo sincronia as mensagens possivelmente irão entrar em conflito e os dispositivos não receberão os dados corretamente.

**Tabela 5:** Tempos relacionados com a transferência de telegramas

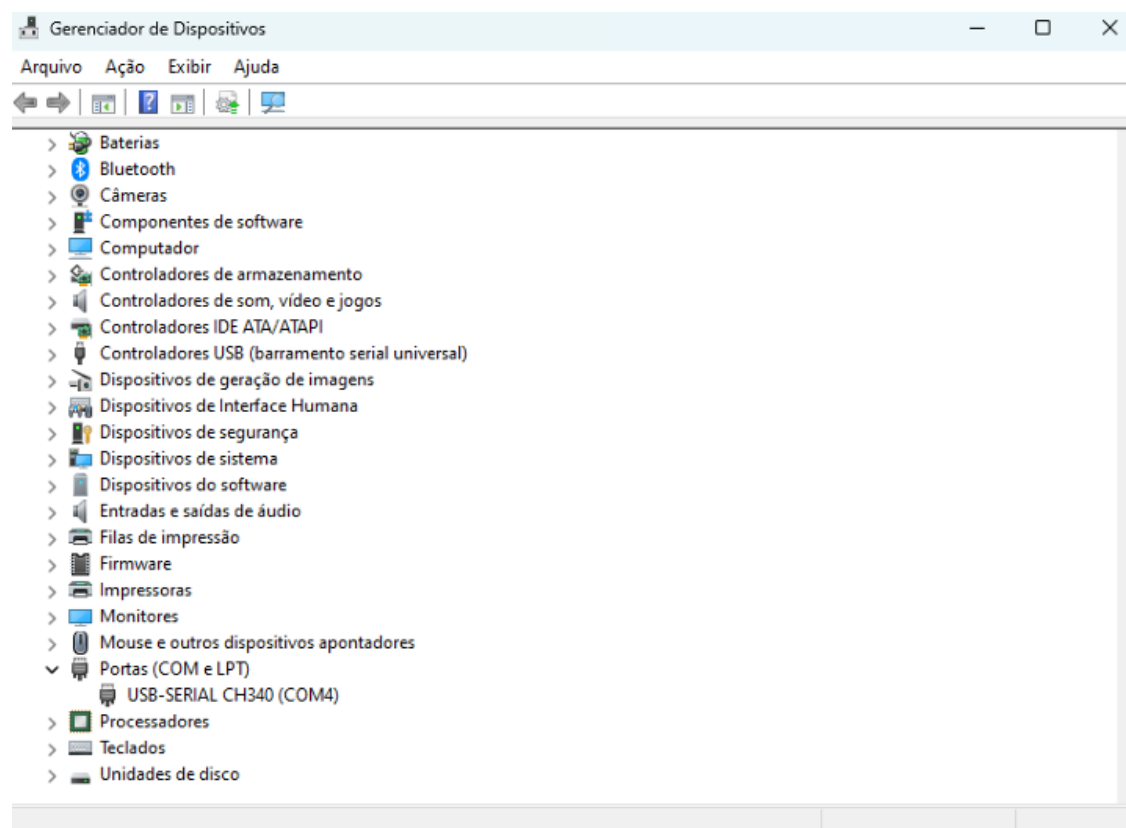
<b>Taxa de Comunicação</b>	<b><math>T_{11bits}</math></b>	<b><math>T_{3,5x}</math></b>
9600 bits/s	1,146 ms	4,010 ms
19200 bits/s	573 $\mu$ s	2,005 ms
38400 bits/s	285 $\mu$ s	1,003 ms

**Fonte:** Autor, 2025

## Apêndice B– Configuração do Conversor Serial na Rede Modbus RTU

Para que a comunicação entre dois ou mais dispositivos numa rede Modbus funcione corretamente, é fundamental que todos os equipamentos envolvidos — como o inversor de frequência, o microcontrolador, o sistema supervisor e o sistema de inteligência — estejam configurados com os mesmos parâmetros de transmissão. Além disso, o conversor USB/Serial RS-485 também deve ser parametrizado de forma compatível. Como o projeto foi desenvolvido num ambiente com sistema operativo Windows 11, apresenta-se de seguida um guia passo-a-passo para a configuração da porta serial neste sistema. Para utilizadores de Linux ou macOS, recomenda-se a consulta de procedimentos específicos para esses sistemas operativos.

O primeiro passo consiste em aceder ao Gestor de Dispositivos do Windows e verificar as portas COM atualmente conectadas ao computador. Caso o conversor USB/Serial RS-485 esteja corretamente ligado e seja reconhecido pelo sistema, será listado como um dispositivo USB-Serial (embora o nome exato possa variar consoante o driver instalado). Um exemplo desta identificação pode ser visualizado na **Fig. B.37**. Se, por outro lado, o dispositivo surgir como “Dispositivo desconhecido”, é sinal de que o driver não está instalado corretamente. Nessa situação, recomenda-se proceder à instalação manual do driver, que pode ser descarregado a partir do site oficial do fabricante.



*Figura 37: Gerenciador de Dispositivos do Windows. Fonte: Autor, 2025*

## ANEXOS

## Anexo A– Códigos de programação

Os códigos desenvolvidos na IDE do Arduino para a leitura dos sensores, bem como o ficheiro do ScadaBR correspondente ao sistema supervisor que permite a sua reconstrução em qualquer máquina, encontram-se disponíveis no link abaixo. Estão igualmente incluídas informações complementares relacionadas com a programação dos dispositivos e documentação de apoio, de forma a facilitar a reprodução e a continuidade do trabalho por outros utilizadores ou investigadores interessados.

- **Repositório do Github:**

<https://github.com/santosarcanjo/IoT-gemios-digitais>

## Anexo B–Inquérito Técnico feito na Austral Cimento Sofala

### Inquérito Técnico sobre Monitoramento Inteligente de Motores Elétricos Críticos na Austral Cimentos de Moçambique

#### Objetivo:

Coletar dados sobre as práticas atuais de monitoramento e manutenção dos motores do forno rotativo e do moinho de cimento, visando propor melhorias através de um sistema baseado em Internet das Coisas (IoT) e Gémeos Digitais.

#### 1. Dados do Respondente

- 1. Nome (opcional): CUSTÓDIO MURROMO
- 2. Cargo/Função: GESTOR DE MANUTENÇÃO
- 3. Área de atuação:  
- ( ) Engenharia elétrica - ☒ Manutenção - ( ) Produção - ( ) Automação - ( ) Outra: \_\_\_\_\_
- 4. Tempo de trabalho na empresa:  
- ( ) Menos de 1 ano - ( ) 1–3 anos - ☒ Mais de 3 anos

#### 2. Motores Críticos na Produção

- 5. Quais motores elétricos são considerados mais críticos no processo de produção? (pode assinalar mais de um)  
- [ ] Forno rotativo - ☒ Moinho de cimento - ☒ Ventiladores industriais - [ ] Transportadores - ☒ Outros: \_\_\_\_\_
- 6. Quais falhas mais comuns ocorrem nesses motores?  
- [ ] Sobreaquecimento - ☒ Vibrações anormais - [ ] Desgaste de rolamentos - [ ] Queima de bobinas - [ ] Outra: \_\_\_\_\_

NOTA: A REBOBINAGEM DE MOTOR ELÉCTRICO TEM CUSTO MÉDIO UM POR ANO NO UNIVERSO DE 200 MOTORES.

### 3. Monitoramento e Manutenção

- 7. Que tipo de manutenção é mais aplicada aos motores críticos?  
- ( ) Corretiva - ☒ Preventiva - ☒ Preditiva
- 8. Existem sensores instalados para monitorar o estado dos motores?  
- ☒ Sim - ( ) Não
- 9. Quais parâmetros são monitorados atualmente?  
- ☒ Temperatura - ☒ Vibração - ☒ Corrente elétrica - [ ] Rotação (RPM) - [ ]  
Outro: VIBRAÇÃO AXIAL, HORIZONTAL E VERTICAL
- 10. Com que frequência os dados são analisados?  
- ☒ Diariamente - ( ) Semanalmente - ( ) Apenas quando há problemas - ( ) Não são analisados  
TAMBÉM HORÁRIA

### 4. Tecnologia e Inovação

- 11. A empresa já utiliza ou está a considerar o uso de tecnologias como IoT ou Gêmeos Digitais?  
- ☒ Sim, já utiliza - ( ) Está em fase de estudo - ( ) Não, mas tem interesse - ( ) Não tem interesse
- 12. Quais seriam, na sua opinião, os principais benefícios de um sistema inteligente de monitoramento com IoT e Gêmeos Digitais?  
CONTROLO COM POUCA INTERVENÇÃO HUMANA.
- 13. Que desafios ou limitações acredita que poderiam dificultar a implementação dessas tecnologias?  
ORGANISMO DIFÍCIL PARA APROX. DE MANUTENÇÃO.

### 5. Comentários Finais

- 14. Deixe aqui qualquer sugestão ou opinião sobre como melhorar o monitoramento dos motores elétricos industriais da fábrica:

NOTA: PONTO Nº9 - EM USO O MACHINERY SENTRY - AVT QUE OBTEM OS DADOS E ENVIAM PARA NUVENS, E PROCESSADO E RETORNADO PODENDO DAR INFORMAÇÃO 1- CRÍTICA, 2- URGENTE FAULT OR 3- OTHER FAULT.