



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



# **Desenvolvimento de um Sistema Supervisório para bancada de estudos do desempenho de células fotovoltaicas**

**Caio Matheus Lopes Ribeiro**

Orientador: Prof. Dr. Andrés Ortiz Salazar

Natal, RN, 17 de janeiro de 2025



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



# **Desenvolvimento de um Sistema Supervisório para bancada de estudos do desempenho de células fotovoltaicas**

**Caio Matheus Lopes Ribeiro**

Orientador: Dr. Andres Ortiz Salazar

**Trabalho de Conclusão de Curso de Graduação** na modalidade Monografia, submetido como parte dos requisitos necessários para conclusão do curso de Engenharia de Computação pela Universidade Federal do Rio Grande do Norte (UFRN/CT).

Natal, RN, 17 de janeiro de 2025

Universidade Federal do Rio Grande do Norte - UFRN  
Sistema de Bibliotecas - SISBI  
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Ribeiro, Caio Matheus Lopes.

Desenvolvimento de um sistema supervisorio para bancada de estudos do desempenho de células fotovoltaicas / Caio Matheus Lopes Ribeiro. - 2025.

52f.: il.

Monografia (Graduação) - Universidade Federal do Rio Grande do Norte, Centro de Tecnologia (CT), Engenharia de Computação, Natal, 2025.

Orientação: Dr. Andres Ortiz Salazar.

1. Energia Fotovoltaica - Monografia. 2. Supervisorio - Monografia. 3. LabVIEW - Monografia. 4. Bancada de Estudos - Monografia. 5. Monitoramento em Tempo Real - Monografia. I. Salazar, Andres Ortiz. II. Título.

RN/UF/BCZM

CDU 004

# **Desenvolvimento de um Sistema Supervisório para bancada de estudos do desempenho de células fotovoltaicas**

**Caio Matheus Lopes Ribeiro**

Monografia aprovada em 17 de janeiro de 2025, pela banca examinadora composta pelos seguintes membros:

---

Prof. Dr. Andres Ortiz Salazar (orientador) ..... DCA/UFRN

---

Dr. Diego Antonio de Moura Fonseca ..... PPgEEC/UFRN

---

Eng. Rafael Augusto de Oliveira Guedes ..... DCA/UFRN

*Aos meus pais, Cezilma Lopes de  
Medeiros e Marcelo Erivaldo  
Ribeiro.*

---

# Agradecimentos

---

Aos meus pais, Cezilma Lopes de Medeiros e Marcelo Erivaldo Ribeiro, por sempre me permitirem seguir o caminho que eu escolhesse, e nunca faltarem com apoio e incentivo em tudo que já realizei na vida.

Aos meus tios, Ana Cecília Lopes de Medeiros Albuquerque e Aldo Teixeira de Albuquerque Neto, por me oferecerem moradia durante meus primeiros semestres de curso, me permitindo ter segurança e conforto para estudar. À minha prima Ilionara Cilanne Lopes de Medeiros por dividir sua casa comigo após a pandemia. Ao meu amigo Gustavo Jerônimo Moura de França e sua família por me acolherem em sua residência quando necessitei.

À minha companheira e meu grande amor Gabriela Cruz Targino, que esteve ao meu lado nas derrotas e conquistas durante a reta final da graduação. Sou eternamente grato por seu imenso apoio companheirismo, nunca permitindo que eu duvidasse de mim mesmo e jamais desistisse dos meus objetivos.

Ao meu orientador, Prof. Dr. Andres Ortiz Salazar, que me apoiou e orientou durante toda minha iniciação científica, contribuindo fundamentalmente para minha formação acadêmica.

À Equipe *Potiguar Rocket Design* pela oportunidade de aprendizado, em especial Mailson Rodrigues de Medeiros Guimarães, responsável por me orientar e me apresentar ao mundo da eletrônica. Agradeço também aos amigos da equipe que compõem o subgrupo *Cringe Móvel*.

Aos amigos do Laboratório de Avaliação e Medição em Petróleo, em especial Adson Emanuel Santos Amaral, que enfrentou todos os desafios da graduação junto a mim desde o dia da matrícula na UFRN, e Rafael Augusto de Oliveira Guedes que, em tão pouco tempo, se tornou um grande amigo e parceiro de estudos.

À UFRN, pela estrutura e oportunidades oferecidas durante toda a graduação.

---

# Resumo

---

A energia solar tem ganhado destaque no Brasil devido à sua grande disponibilidade e ao crescente interesse por fontes de energia renovável. O país, com sua vasta incidência solar ao longo do ano, apresenta condições ideais para o aproveitamento dessa forma de energia, contribuindo para a diversificação da matriz energética e a redução de impactos ambientais. Diante desse contexto, este trabalho apresenta o desenvolvimento de um sistema supervisorio para monitoramento e controle de uma bancada de estudos em energia fotovoltaica, localizada no Laboratório de Avaliação de Medição em Petróleo (LAMP). O sistema foi implementado utilizando a plataforma LabVIEW, que permite a aquisição e análise de dados em tempo real, garantindo uma supervisão eficiente dos parâmetros críticos, como corrente, tensão, irradiância solar, temperatura do módulo fotovoltaico e do ambiente, armazenando os dados coletados. O sistema supervisorio oferece ao usuário a possibilidade de monitorar a bancada em tempo real, possibilitando a detecção de falhas e anomalias durante testes.

**Palavras-chave:** Energia Fotovoltaica, Supervisorio, LabVIEW, Bancada de Estudos, Monitoramento em Tempo Real.

---

# Abstract

---

Solar energy has gained prominence in Brazil due to its abundant availability and the growing interest in renewable energy sources. With its vast year-round solar incidence, the country presents ideal conditions for harnessing this form of energy, contributing to the diversification of the energy matrix and the reduction of environmental impacts. In this context, this work presents the development of a supervisory system for monitoring and controlling a photovoltaic energy study bench, located in the Laboratório de Avaliação e Medição de Petróleo (LAMP). The system was implemented using the LabVIEW platform, which enables real-time data acquisition and analysis, ensuring efficient supervision of critical parameters such as current, voltage, solar irradiance, the temperature of the photovoltaic module and the environment, while storing the collected data. The supervisory system provides users with the capability to monitor the bench in real-time, allowing the detection of faults and anomalies during tests.

**Keywords:** Photovoltaic Energy, Supervisory System, LabVIEW, Experimental Bench, Real-Time Monitoring.

---

# Sumário

---

|  |            |
|--|------------|
| <b>Sumário</b>   | <b>i</b>   |
| <b>Lista de Figuras</b>  | <b>iii</b> |
| <b>Lista de Tabelas</b>  | <b>v</b>   |
| <b>1 Introdução</b>  | <b>1</b>   |
| <b>2 Problema</b>  | <b>2</b>   |
| 2.1 Bancada de Estudos . . . . .                                 | 2          |
| 2.2 Objetivos . . . . .  | 3          |
| 2.2.1 Monitoramento de temperaturas . . . . .                    | 3          |
| 2.2.2 Monitoramento de tensão e corrente . . . . .               | 3          |
| 2.2.3 Controle e monitoramento de luminosidade . . . . .         | 3          |
| 2.2.4 Armazenamento de dados . . . . .                           | 4          |
| <b>3 Fundamentação Teórica</b>                                   | <b>5</b>   |
| 3.1 Sistemas fotovoltaicos . . . . .                             | 5          |
| 3.2 Sistema supervisorio . . . . .                               | 6          |
| 3.3 Ponto de máxima potência . . . . .                           | 7          |
| 3.4 Comunicação I2C . . . . .                                    | 8          |
| <b>4 Metodologia</b>   | <b>10</b>  |
| 4.1 Materiais . . . . .  | 10         |
| 4.1.1 INA219 . . . . .   | 10         |
| 4.1.2 BH1750 . . . . .   | 11         |
| 4.1.3 Sensores de Temperatura: LM35 e LM335 . . . . .            | 12         |
| 4.1.4 Módulo com 4 Relés . . . . .                               | 12         |
| 4.1.5 NI USB6008 . . . . .                                       | 13         |
| 4.1.6 Arduino UNO . . . . .                                      | 15         |
| 4.2 Arquitetura e funcionamento do Sistema . . . . .             | 16         |
| <b>5 Implementação</b>   | <b>18</b>  |
| 5.1 Implementação do supervisorio no LabVIEW . . . . .           | 18         |
| 5.1.1 Controle de luminosidade . . . . .                         | 19         |
| 5.1.2 Monitoramento de tensão, corrente e luminosidade . . . . . | 23         |
| 5.1.3 Monitoramento de temperaturas . . . . .                    | 24         |

|          |  |           |
|----------|--|-----------|
| 5.1.4    | Armazenamento de dados . . . . .                           | 26        |
| <b>6</b> | <b>Experimentos e Resultados</b>                           | <b>28</b> |
| 6.1      | Controle de luminosidade . . . . .                         | 28        |
| 6.2      | Monitoramento de tensão, corrente e luminosidade . . . . . | 29        |
| 6.3      | Monitoramento de temperaturas . . . . .                    | 30        |
| 6.4      | Armazenamento de dados . . . . .                           | 32        |
| <b>7</b> | <b>Conclusão</b>   | <b>33</b> |
| 7.1      | Trabalhos futuros . . . . .                                | 33        |
|          | <b>Referências bibliográficas</b>                          | <b>35</b> |
| <b>A</b> | <b>Informações adicionais</b>                              | <b>39</b> |
| A.1      | Algoritmo Arduino . . . . .                                | 39        |
| A.2      | Código completo LabVIEW . . . . .                          | 40        |

---

# Lista de Figuras

---

|      |  |    |
|------|--|----|
| 2.1  | Bancada de estudos. . . . .  | 2  |
| 3.1  | Mapa de irradiação do Brasil. . . . .                                      | 6  |
| 3.2  | Sistema supervisorio de um poço automatizado com bombeio mecânico .        | 7  |
| 3.3  | Curvas I-V e P-V de um módulo ou conjunto de módulos fotovoltaicos. .      | 8  |
| 3.4  | Comunicação I2C entre controlador e periférico. . . . .                    | 9  |
| 4.1  | Sensor de corrente e tensão INA219. . . . .                                | 10 |
| 4.2  | Sensor de corrente e tensão BH1750. . . . .                                | 11 |
| 4.3  | Sensor de corrente e tensão INA219. . . . .                                | 12 |
| 4.4  | Módulo de 4 relés. . . . .   | 13 |
| 4.5  | Módulo de aquisição de dados NI USB6008. . . . .                           | 14 |
| 4.6  | Pinos do DAQ USB6008. . . . .  | 14 |
| 4.7  | Arduino UNO. . . . .   | 16 |
| 4.8  | Arquitetura do sistema. . . . .  | 17 |
| 5.1  | Interface do sistema supervisorio. . . . .                                 | 19 |
| 5.2  | Módulo de gráficos do sistema supervisorio. . . . .                        | 19 |
| 5.3  | Módulo de controle de luminosidade. . . . .                                | 20 |
| 5.4  | Configuração dos pinos de saída USB6008. . . . .                           | 21 |
| 5.5  | Módulo 4 relés conectado às lâmpadas. . . . .                              | 21 |
| 5.6  | Montagem de um circuito de lâmpadas acionadas por relé. . . . .            | 22 |
| 5.7  | Lógica implementada em LabVIEW para o acionamento das lâmpadas. .          | 22 |
| 5.8  | Configuração de porta e leitura dos dados na tela do supervisorio. . . . . | 23 |
| 5.9  | Lógica implementada em LabVIEW para a leitura do Arduino. . . . .          | 24 |
| 5.10 | Apresentação das temperaturas na tela do supervisorio. . . . .             | 25 |
| 5.11 | Configuração dos pinos de entrada analógica USB6008. . . . .               | 25 |
| 5.12 | Lógica implementada em LabVIEW para o monitoramento de temperatura.        | 26 |
| 5.13 | Caminho do arquivo na tela do supervisorio. . . . .                        | 26 |
| 5.14 | Código do armazenamento de dados . . . . .                                 | 27 |
| 6.1  | Sistema acionado com apenas 50% de luminosidade total. . . . .             | 28 |
| 6.2  | Sistema acionado com 100% de luminosidade. . . . .                         | 29 |
| 6.3  | Montagem do circuito teste com o LED. . . . .                              | 30 |
| 6.4  | Teste de monitoramento do INA219 e BH1750. . . . .                         | 30 |
| 6.5  | Monitoramento de temperaturas com as 4 lâmpadas ligadas. . . . .           | 31 |
| 6.6  | Foto dos dados salvos formato .xlsx. . . . .                               | 32 |

A.1 Algoritmo completo em LabVIEW. . . . . 40

---

# Lista de Tabelas

---

|     |  |    |
|-----|--|----|
| 4.1 | Especificações do INA219 . . . . .               | 11 |
| 4.2 | Especificações do BH1750 . . . . .               | 11 |
| 4.3 | Especificações do LM35 . . . . .                 | 12 |
| 4.4 | Especificações do LM335 . . . . .                | 13 |
| 4.5 | Especificações do Módulo com 4 Relés . . . . .   | 13 |
| 4.6 | Especificações Técnicas do NI USB-6008 . . . . . | 15 |
| 4.7 | Especificações Técnicas do Arduino Uno . . . . . | 16 |

---

# Capítulo 1

## Introdução

---

Nos últimos anos, a crescente preocupação com a sustentabilidade, aliada aos avanços tecnológicos e à redução dos custos dos equipamentos, tem impulsionado os investimentos em tecnologias renováveis. O Brasil possui aproximadamente 50% de sua matriz energética proveniente de energias sustentáveis, sendo a energia solar uma das maiores contribuintes (Governo do Brasil 2021). A adoção de sistemas fotovoltaicos em usinas, residências e comércios cresce devido à praticidade e ao baixo custo de implementação e manutenção em comparação com hidrelétricas e eólicas.

Com a crescente demanda por energia limpa, pesquisas e desenvolvimento contínuos são essenciais para aprimorar os equipamentos e tecnologias, buscando reduzir perdas e aumentar a eficiência, tornando a energia solar mais viável e sustentável. Estudos e testes em ambientes naturais são frequentemente inviáveis devido às condições variáveis, por isso, bancadas experimentais oferecem um ambiente controlado para o desenvolvimento prático do conhecimento teórico.

Nesse contexto, um sistema supervisorio para monitorar bancadas de estudos fotovoltaicos torna-se um aliado dos pesquisadores, permitindo o acompanhamento em tempo real de dados como tensão, corrente e temperatura, facilitando a identificação de falhas e a análise de desempenho. Isso melhora a precisão dos experimentos e agiliza a tomada de decisão, além de proporcionar uma interface amigável para pesquisadores e estudantes, promovendo uma melhor compreensão dos fenômenos.

Este trabalho apresenta o desenvolvimento de um sistema supervisorio projetado para uma bancada de estudos de sistemas fotovoltaicos no Laboratório de Avaliação de Medição em Petróleo (LAMP) da UFRN. Utilizando o software *Laboratory Virtual Instrument Engineering Workbench* (LabVIEW), o sistema monitora e registra dados em tempo real, como tensão, corrente e temperatura de uma célula fotovoltaica, e controla lâmpadas que simulam a irradiação solar, facilitando a análise e otimização dos experimentos.

---

# Capítulo 2

## Problema

---

### 2.1 Bancada de Estudos

O supervisor desenvolvido neste projeto tem como objetivo monitorar uma bancada de estudo de desempenho de células fotovoltaicas desenvolvida pela equipe do LAMP, apresentada no "21th International Conference on Renewable Energies and Power Quality (ICREPQ'23)".

Essa bancada, ilustrada na Figura 2.1, é composta por um painel fotovoltaico, lâmpadas que simulam a iluminação solar, células Peltier que resfriam a placa, uma interface homem-máquina para operações manuais e um tanque de água para resfriar as células Peltier. A bancada é fundamental para a pesquisa e desenvolvimento no campo fotovoltaico, possibilitando um estudo detalhado e sistemático das condições que afetam a eficiência dos sistemas solares, e o sistema supervisor oferece maior apoio e praticidade para o desenvolvimento de estudos.

Figura 2.1: Bancada de estudos.



Fonte: Autoria própria, 2024.

## 2.2 Objetivos

O principal objetivo é desenvolver um supervisor capaz de atender às seguintes exigências: monitoramento de duas temperaturas diferentes, monitoramento de tensão e corrente, monitoramento de luminosidade e armazenamento de dados.

### 2.2.1 Monitoramento de temperaturas

O monitoramento das temperaturas da placa e do ambiente é necessário por diversos fatores. Um deles é a avaliação da eficiência da célula fotovoltaica, tendo em vista que seu desempenho tende a cair com o aumento da temperatura da placa (Salazar et al. 2023). Outro fator é a comparação do desempenho da célula sob diversas condições de ambiente, oferecendo um contexto para o entendimento da influência de condições externas na eficiência do sistema fotovoltaico.

### 2.2.2 Monitoramento de tensão e corrente

Monitorar a tensão e a corrente em um sistema fotovoltaico é essencial para garantir seu funcionamento eficiente e seguro. A tensão e a corrente fornecem informações cruciais sobre o desempenho do sistema, permitindo determinar o ponto de máxima potência para maximizar a energia gerada. Além disso, variações anormais nesses parâmetros podem indicar falhas, como problemas nos módulos, sombreamento parcial, conexões soltas ou curto-circuito, possibilitando a detecção prévia de problemas. O monitoramento também permite avaliar a eficiência do sistema ao comparar a energia gerada com a irradiação solar e facilita a manutenção preditiva ao identificar tendências de degradação.

### 2.2.3 Controle e monitoramento de luminosidade

Executar o controle de luminosidade, especialmente no contexto da bancadas de estudos, permite a simulação de diferentes condições de iluminação, possibilitando a realização de testes controlados e a análise do desempenho dos módulos sob variadas intensidades de luz. Isso é fundamental para validar o comportamento do sistema em situações que simulam a operação real e para aprimorar o desenvolvimento de tecnologias fotovoltaicas.

O monitoramento de luminosidade, por sua vez, é igualmente importante, pois a quantidade de luz solar incidente afeta diretamente a energia gerada pelos módulos fotovoltaicos. A correlação entre a luminosidade e o desempenho do sistema ajuda a identificar situações de sombreamento temporário e a ajustar o sistema para otimizar a geração de energia em diferentes condições de luz.

### **2.2.4 Armazenamento de dados**

Possuir um sistema de armazenamento dos dados de testes permite a análise histórica do desempenho do sistema, possibilitando a identificação de padrões de operação e tendências que podem indicar falhas iminentes ou a necessidade de manutenção. Além disso, a análise dos dados armazenados ajuda a otimizar o sistema ao longo do tempo, ajustando parâmetros para maximizar a eficiência energética. Os dados armazenados também são fundamentais para fins de pesquisa e desenvolvimento, permitindo que os pesquisadores revisem e comparem resultados de diferentes condições de teste.

---

# Capítulo 3

## Fundamentação Teórica

---

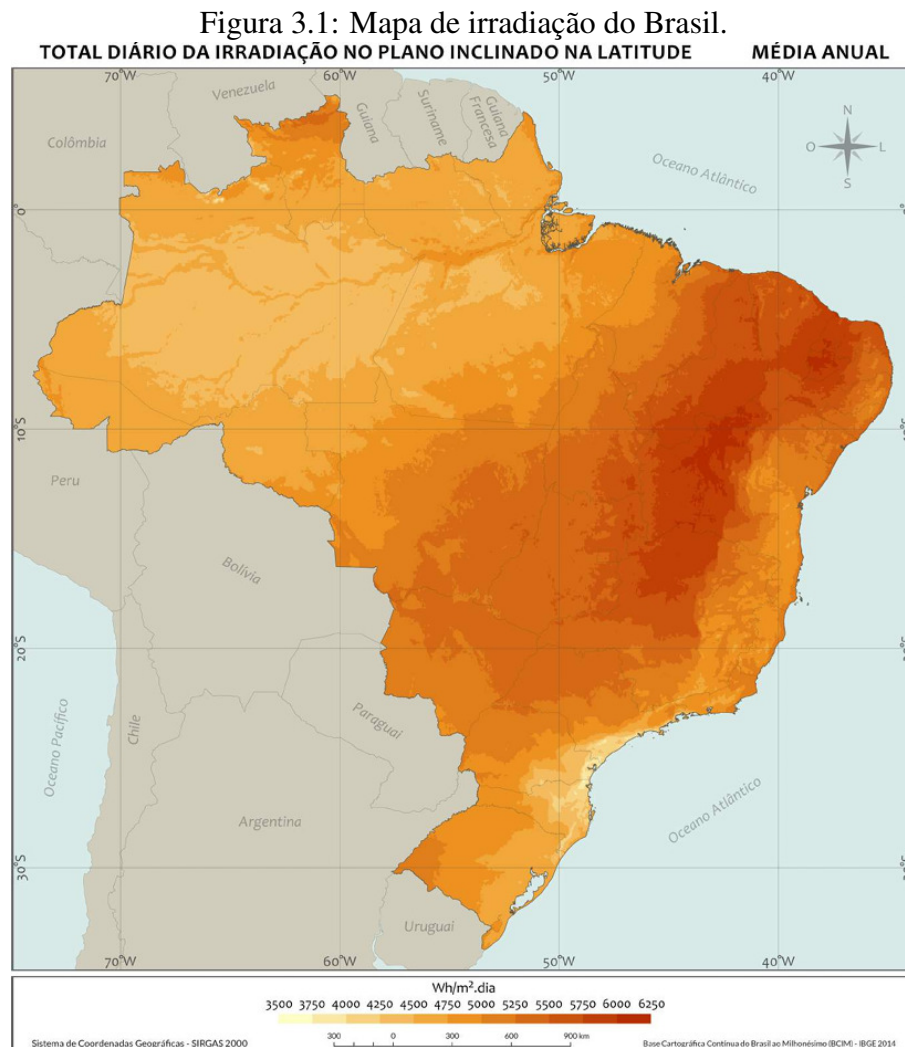
Este capítulo apresenta os fundamentos teóricos que sustentam este estudo, abordando temas cruciais para o entendimento do sistema proposto. São explorados os seguintes tópicos: sistemas fotovoltaicos, que descrevem a conversão de energia solar em elétrica; sistemas supervisórios, essenciais para o monitoramento e controle de processos; o ponto de máxima potência, conceito fundamental para otimização da eficiência energética em células fotovoltaicas; e o protocolo de comunicação I2C, utilizado para a interação eficiente entre componentes eletrônicos.

### 3.1 Sistemas fotovoltaicos

O Brasil possui uma das maiores potências solares do mundo, devido à sua localização geográfica privilegiada próxima à linha do Equador. De acordo com o Atlas Brasileiro de Energia Solar (2017), a irradiação solar no Brasil varia entre 4.500 a 6.300 Wh/m<sup>2</sup> por dia, valores que superam a média de muitos países europeus, onde a energia solar já é amplamente utilizada. Essa abundância de recurso solar torna o Brasil um local estratégico para o desenvolvimento e expansão de sistemas fotovoltaicos. A Figura 3.1 exibe o mapa da média anual de irradiação no Brasil.

De modo geral, os sistemas fotovoltaicos consistem em módulos que captam a luz solar e a convertem em energia elétrica através do efeito fotovoltaico. Esta eletricidade pode ser usada imediatamente, armazenada em baterias para uso posterior, ou ainda injetada na rede elétrica. Os principais componentes de um sistema desse tipo são: módulos fotovoltaicos, inversores, estruturas de suporte, e, em alguns casos, sistemas de armazenamento e monitoramento.

O uso do método de captação de energia solar como energia oferece diversos benefícios, dentre eles destacam-se: a sua longa vida útil, a pouca necessidade de manutenção, não produzem poluição ou ruído e seu combustível é a luz solar natural. Além disso, os sistemas são modulares ou expansíveis, leves, fáceis de transportar e estão disponíveis ao longo de todo o ano (Marini & Rossi 2002).



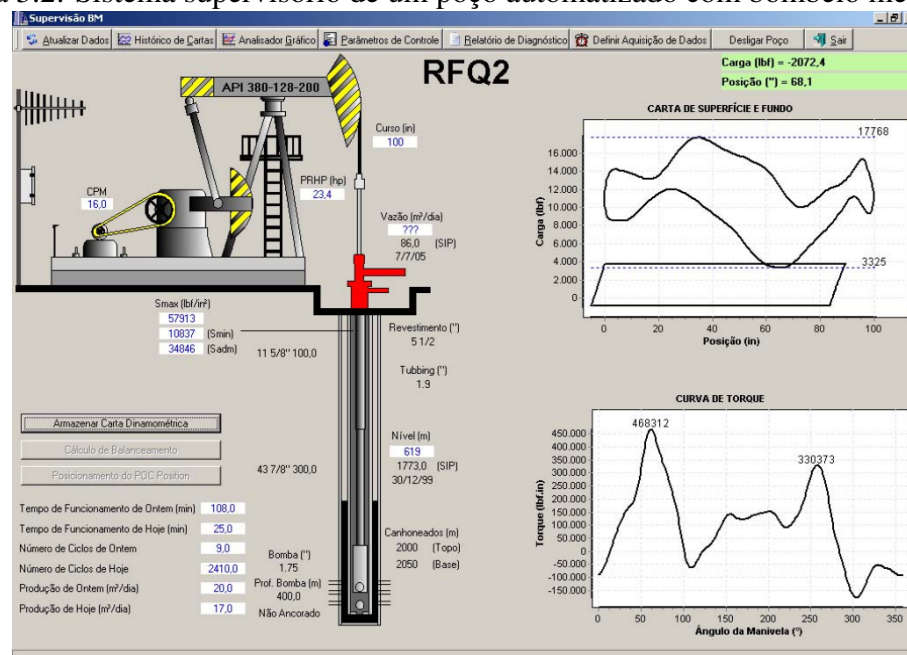
Fonte: INPE, 2017

## 3.2 Sistema supervisório

Os sistemas supervisórios desempenham um papel crucial na automação industrial, proporcionando monitoramento, controle e gerenciamento de processos produtivos em tempo real. Segundo Marcondes (2005), um sistema supervisório é composto por ferramentas de software capazes de coletar, processar e apresentar dados provenientes de dispositivos de controle e sensores para os operadores e gestores. Essas plataformas não apenas monitoram variáveis do processo, mas também permitem intervenções manuais ou automáticas no sistema, como o supervisório apresentado na Figura 3.2 que trata de uma aplicação na indústria do petróleo, presente no artigo de Souza et al. (2006).

Dessa forma, um supervisório comumente possui uma interface amigável e de fácil interpretação para o operador, proporcionando uma visualização clara e em tempo real dos dados. Outra característica essencial para um supervisório é a capacidade de alertar através de atuadores sonoros e/ou visuais para notificar falhas ou desvios nos parâmetros

Figura 3.2: Sistema supervisorio de um poço automatizado com bombeio mecânico .



Fonte: Souza et al., 2006.

em um determinado sistema, além de permitir o controle remoto e a configuração dos dispositivos do sistema.

O uso de sistemas assim é comum na indústria, com o objetivo de oferecer uma visão centralizada dos processos, integrando informações provenientes de sensores, controladores e outros dispositivos de campo. Essa integração possibilita o monitoramento de variáveis críticas, como temperatura, pressão, fluxo e potência, e a execução de ações corretivas automatizadas ou manuais, conforme necessário. Quando a supervisão e o controle de um processo são realizados por meio de um sistema supervisorio, o processamento das variáveis de campo torna-se mais ágil e eficiente (COELHO 2010).

### 3.3 Ponto de máxima potência

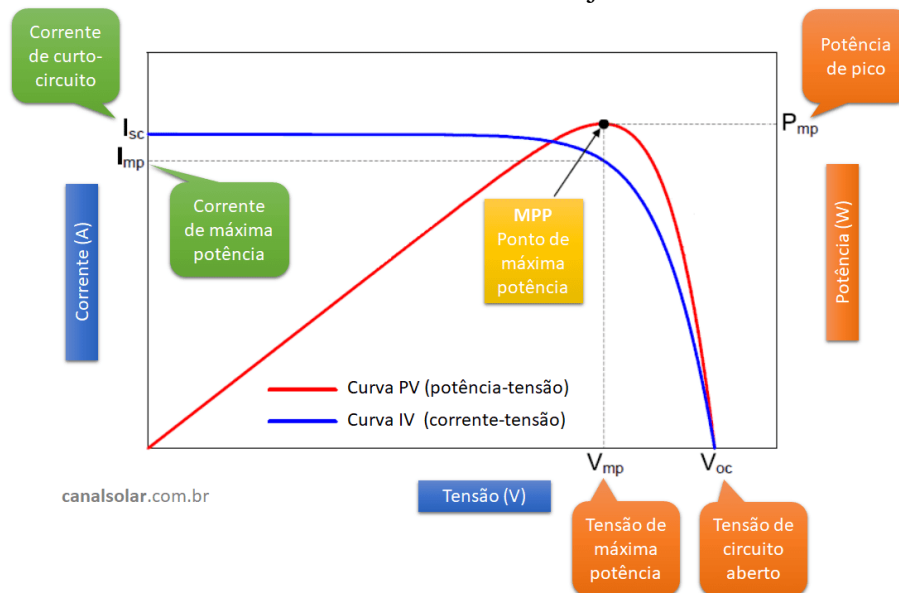
O conceito de ponto de máxima potência (MPP, do inglês *Maximum Power Point*) é crucial em sistemas de energia solar fotovoltaica, estando diretamente relacionado à otimização da conversão de energia. As células solares funcionam convertendo a luz solar em eletricidade por meio do efeito fotovoltaico. A energia gerada por um painel solar depende de variáveis como a tensão ( $V$ ) e a corrente elétrica ( $I$ ), sendo a potência elétrica expressa pela equação:

$$P = V \cdot I$$

A relação entre a tensão e a corrente em uma célula solar não é linear, resultando em uma curva característica  $I \cdot V$ , como pode ser observado na Figura 3.3. Nessa curva, existe

um ponto específico em que o produto  $V \cdot I$  atinge seu valor máximo, o MPP.

Figura 3.3: Curvas I-V e P-V de um módulo ou conjunto de módulos fotovoltaicos.



Fonte: Canal Solar, 2024.

A análise das curvas I-V (corrente e tensão) e P-V (potência e tensão) é essencial para o dimensionamento e otimização de sistemas fotovoltaicos, onde a curva I-V indica a eficiência da célula em converter luz solar em corrente elétrica, enquanto a curva P-V ajuda a identificar o MPP.

O Rastreamento do Ponto de Máxima Potência (MPPT, do inglês *Maximum Power Point Tracker*), por sua vez, é uma técnica amplamente utilizada em sistemas fotovoltaicos para otimizar a conversão de energia elétrica da placa fotovoltaica. Segundo Abdelaziz & Almoataz (2020), o MPPT atua ajustando continuamente as condições de operação do sistema fotovoltaico de forma a maximizar a potência extraída do painel solar, mesmo diante de variações de irradiância e temperatura.

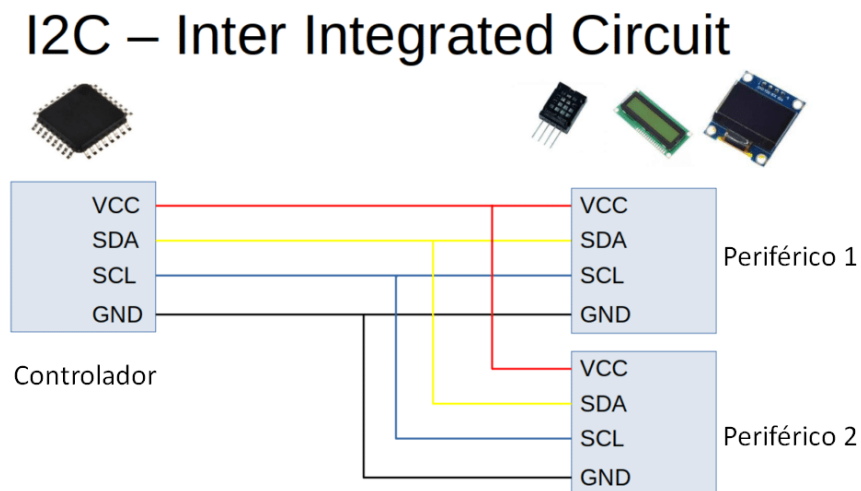
### 3.4 Comunicação I2C

A comunicação I2C (*Inter-Integrated Circuit*) é um protocolo serial síncrono amplamente utilizado para comunicação entre dispositivos integrados em sistemas embarcados. Desenvolvido pela Philips Semiconductor na década de 1980, o I2C é projetado para permitir que múltiplos dispositivos, como microcontroladores, sensores e periféricos, compartilhem um mesmo barramento de comunicação, utilizando apenas duas linhas: *SDA* (*Serial Data Line*) para a transmissão de dados e *SCL* (*Serial Clock Line*) para o sincronismo.

Esse protocolo opera na configuração controlador-periférico (ou mestre-escravo), no qual um dispositivo controlador administra o barramento e gerencia a comunicação, enquanto os dispositivos periféricos respondem às solicitações do controlador, como é mos-

trado na Figura 3.4. Ele permite a conexão de múltiplos dispositivos no mesmo barramento, diferenciados por endereços únicos de 7 ou 10 *bits*. O protocolo utiliza mensagens estruturadas em pacotes contendo *bits* de início (*start*), endereço do periférico, dados e *bits* de parada (*stop*). Além disso, esse método de comunicação suporta diferentes modos de velocidade, como *Standard Mode* (100 kbps), *Fast Mode* (400 kbps) e *High-Speed Mode* (até 3,4 Mbps) (Mankar et al. 2014). Utiliza também um sistema de *acknowledge* para confirmar o recebimento de dados, garantindo a integridade da comunicação.

Figura 3.4: Comunicação I2C entre controlador e periférico.



Fonte: Adaptado de Wantronics, 2024.

No início de uma comunicação, o controlador emite um sinal de início (*start condition*), seguido pelo endereço do dispositivo periférico com o qual deseja se comunicar. O periférico correspondente responde com um sinal de reconhecimento (*acknowledge*), e a transferência de dados é iniciada. Após a transmissão, o controlador emite um sinal de parada (*stop condition*), indicando o término da comunicação.

---

# Capítulo 4

## Metodologia

---

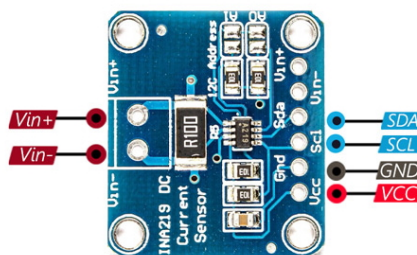
### 4.1 Materiais

Para a implementação do sistema, diversos materiais e componentes foram utilizados, abrangendo desde sensores específicos até módulos de controle e análise. A seguir, serão apresentados os principais componentes empregados, destacando suas funcionalidades, características técnicas e a importância de sua escolha para a realização do projeto.

#### 4.1.1 INA219

O sensor INA219 (Figura 4.1) é um sensor de corrente, tensão e potência de alta precisão que permite a medição da corrente que flui através de um resistor shunt externo, bem como a tensão do sistema.

Figura 4.1: Sensor de corrente e tensão INA219.



Fonte: Arduino e cia., 2024.

Ele utiliza comunicação I2C para enviar os dados para um microcontrolador, e é um sensor barato, de fácil acesso e com boa faixa de valores para se trabalhar, o que o torna

ideal para monitoramento de consumo de energia em diversos dispositivos. A Tabela 4.1 apresenta as especificações técnicas do INA219.

Tabela 4.1: Especificações do INA219

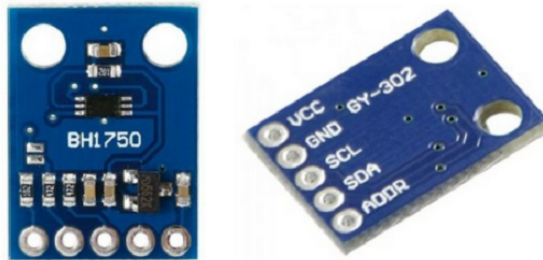
| Parâmetro                     | Especificação |
|-------------------------------|---------------|
| Tensão de alimentação         | 3V a 5.5V     |
| Faixa de corrente             | $\pm 3.2A$    |
| Resolução de corrente         | 0.1mA         |
| Faixa de tensão de barramento | 0V a 26V      |
| Comunicação                   | I2C           |
| Precisão                      | 1%            |

Fonte: Adafruit, 2024.

### 4.1.2 BH1750

O BH1750, presente na Figura 4.2 é um sensor digital de luz ambiente que oferece alta precisão e uma interface I2C para fácil comunicação com microcontroladores. Ele é amplamente utilizado para medir a luminosidade em ambientes internos e externos, devido a sua faixa de medição de luminosidade que pode ser observada na Tabela 4.2, sendo útil em projetos que exigem controle baseado em iluminação, como sistemas de iluminação automática.

Figura 4.2: Sensor de corrente e tensão BH1750.



Fonte: Erriez, 2025.

Tabela 4.2: Especificações do BH1750

| Parâmetro               | Especificação |
|-------------------------|---------------|
| Tensão de alimentação   | 2.4V a 3.6V   |
| Faixa de medição de luz | 1 a 65535 lx  |
| Precisão                | 20%           |
| Consumo de corrente     | 0.12mA        |
| Comunicação             | I2C           |

Fonte: Adafruit, 2025.



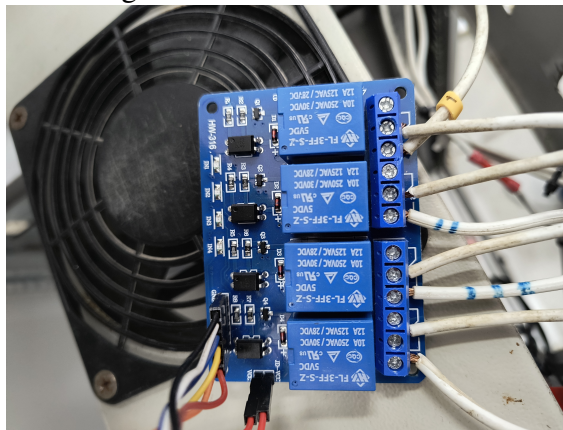
Tabela 4.4: Especificações do LM335

| Parâmetro                        | Especificação         |
|----------------------------------|-----------------------|
| Tensão de alimentação            | 4V a 30V              |
| Faixa de temperatura de operação | -40°C a 100°C         |
| Precisão típica                  | $\pm 1^\circ\text{C}$ |
| Sensibilidade                    | 10mV/°C               |

Fonte: Texas Instruments, 2025.

dispositivos de alta potência com sinais de baixa potência. Suas especificações técnicas são apresentadas na Figura 4.5.

Figura 4.4: Módulo de 4 relés.



Fonte: Autoria própria, 2025.

Tabela 4.5: Especificações do Módulo com 4 Relés

| Parâmetro             | Especificação                 |
|-----------------------|-------------------------------|
| Tensão de alimentação | 5V DC                         |
| Corrente por relé     | 10A a 250V AC ou 10A a 30V DC |
| Tempo de operação     | 10ms                          |
| Tempo de liberação    | 5ms                           |
| Isolamento            | Opto-isolado                  |

Fonte: Alldatasheet, 2024.

#### 4.1.5 NI USB6008

Com o objetivo de integrar todos os sensores e componentes do sistema, e devido à sua praticidade e fácil manuseio, o módulo de aquisição de dados DAQ (*Data Acquisition*) NI-USB6008 (Figura 4.5) foi escolhido para realizar a aquisição de dados.

Entre suas principais características, destacam-se as 8 entradas analógicas, configuráveis como diferenciais ou de terminação única, com resolução de 12 *bits* e taxa de amostragem de até 10 kS/s, além de 2 saídas analógicas também com 12 *bits* de resolução. O

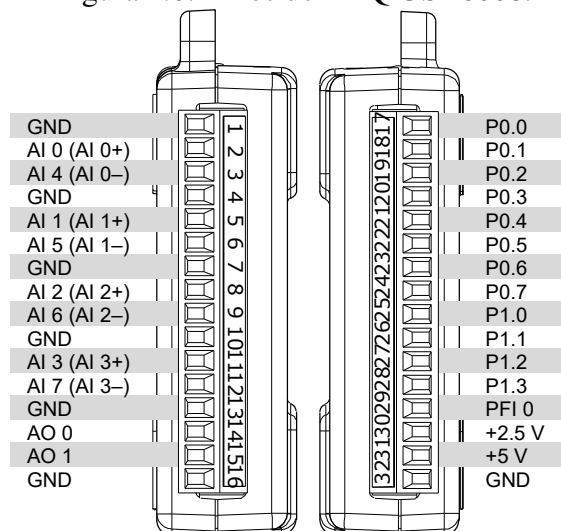
Figura 4.5: Módulo de aquisição de dados NI USB6008.



Fonte: Digikey, 2024.

módulo conta com 12 canais digitais configuráveis como entrada ou saída, permitindo o controle de dispositivos externos, e um contador de 32 *bits* para medições de eventos e pulsos (National Instruments 2018). Alimentado diretamente pela porta USB, o NI USB-6008 dispensa a necessidade de fontes externas de energia. A distribuição de pinos do módulo DAQ é apresentada na Figura 4.6.

Figura 4.6: Pinos do DAQ USB6008.



Fonte: National Instruments, 2024.

A praticidade desse dispositivo é ideal para tarefas como medições de sensores, aquisição de dados em experimentos científicos, monitoramento de sinais e prototipagem de sistemas. Sua portabilidade, eficiência e fácil integração com o software LabVIEW fo-

ram cruciais para sua escolha neste projeto. As especificações técnicas do módulo de aquisição de dados são descritas na Tabela 4.6.

Tabela 4.6: Especificações Técnicas do NI USB-6008

| <b>Característica</b>                 | <b>Especificação</b>   |
|---------------------------------------|--|
| <b>Número de Entradas Analógicas</b>  | 8 (Single-ended) ou 4 (Diferenciais)   |
| <b>Resolução</b>                      | 12 bits  |
| <b>Taxa de Amostragem</b>             | 10 kS/s (Taxa Máxima de Amostragem)  |
| <b>Intervalo de Entrada Analógica</b> | $\pm 10 \text{ V}$ , $\pm 5 \text{ V}$ , $\pm 2 \text{ V}$ , $\pm 1 \text{ V}$ |
| <b>Número de Saídas Analógicas</b>    | 2  |
| <b>Faixa de Saída Analógica</b>       | 0 a 5 V  |
| <b>Resolução de Saída Analógica</b>   | 12 bits  |
| <b>Número de Entradas Digitais</b>    | 12   |
| <b>Número de Saídas Digitais</b>      | 12   |
| <b>Tensão de Operação Digital</b>     | 0 a 5 V  |
| <b>Contadores/Temporizadores</b>      | 1 contador de 32 bits  |
| <b>Comunicação</b>                    | USB 2.0  |
| <b>Dimensões</b>                      | 88,9 mm x 62,2 mm x 23,1 mm  |
| <b>Peso</b>                           | Aproximadamente 85 g   |
| <b>Fonte de Alimentação</b>           | Alimentado via USB   |

Fonte: National Instruments, 2018.

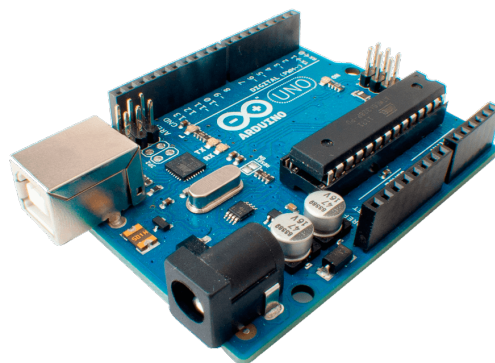
Apesar das inúmeras vantagens, o módulo da National USB6008 possui uma limitação quanto à comunicação com sensores do tipo I2C. Sendo assim, optou-se pela utilização de um Arduino Uno para auxiliar no sistema.

#### 4.1.6 Arduino UNO

O Arduino Uno apresentado na Figura 4.7 é uma das placas mais conhecidas da plataforma Arduino, amplamente utilizada em projetos de prototipagem eletrônica e automação. Baseado no microcontrolador ATmega328P, o Arduino Uno possui 14 pinos digitais, 6 pinos analógicos e recursos como comunicação serial via USB, permitindo fácil interação com um computador para programação e monitoramento. Sua popularidade se deve à simplicidade de uso, grande comunidade de desenvolvedores e à vasta documentação disponível, o que facilita a criação de projetos para iniciantes e especialistas.

Uma das características mais vantajosas do Arduino Uno é sua capacidade de comunicação I2C, um protocolo de comunicação serial que permite a conexão de múltiplos dispositivos usando apenas dois pinos: SDA (dados) e SCL (clock) (STA Eletrônica 2025). Esse protocolo facilita a comunicação entre o Arduino e uma grande variedade de sensores, como o INA219 e o BH1750, reduzindo a quantidade de fios e simplificando o design do circuito. O suporte ao I2C torna o Arduino Uno ideal para projetos que envolvem o uso de vários sensores e módulos, proporcionando uma integração eficiente e de baixo custo com dispositivos externos. Suas especificações são descritas na Tabela 4.7 a seguir:

Figura 4.7: Arduino UNO.



Fonte: D&D Componentes, 2024.

Tabela 4.7: Especificações Técnicas do Arduino Uno

| <b>Característica</b>                  | <b>Especificação</b> |
|--|----------------------|
| <b>Microcontrolador</b>                | ATmega328P           |
| <b>Tensão de Operação</b>              | 5 V                  |
| <b>Tensão de Entrada (recomendada)</b> | 7-12 V               |
| <b>Entradas Digitais/Analógicas</b>    | 14 (6 PWM) / 6       |
| <b>Corrente DC por I/O Pin</b>         | 40 mA                |
| <b>Corrente DC para 3.3V Pin</b>       | 50 mA                |
| <b>Memória Flash</b>                   | 32 KB (ATmega328P)   |
| <b>SRAM</b>                            | 2 KB (ATmega328P)    |
| <b>EEPROM</b>                          | 1 KB (ATmega328P)    |
| <b>Velocidade do Clock</b>             | 16 MHz               |
| <b>Dimensões</b>                       | 68,6 mm x 53,4 mm    |
| <b>Peso</b>                            | 25 g                 |

Fonte: STA Eletrônica, 2024.

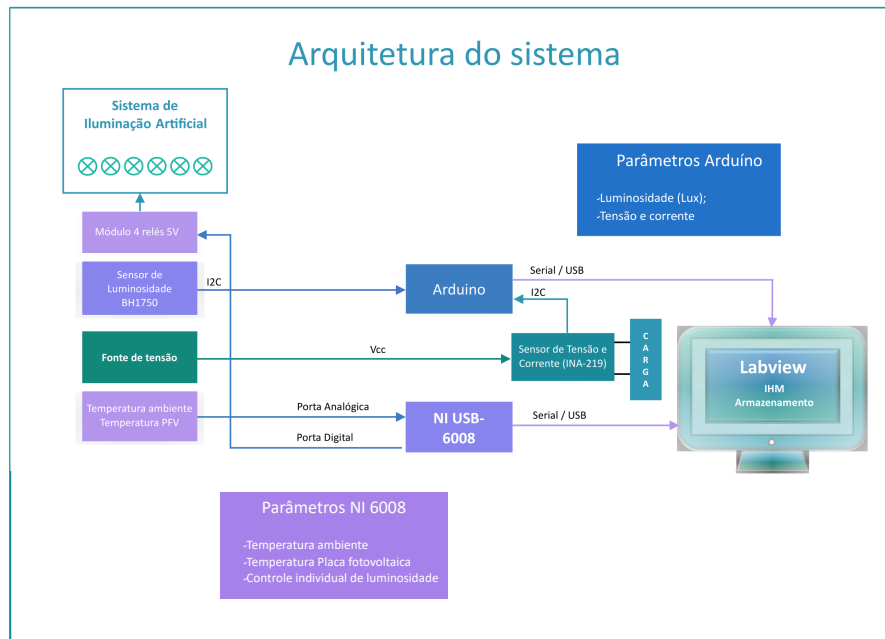
## 4.2 Arquitetura e funcionamento do Sistema

Conforme mostrado na Figura 4.8, no coração do sistema está o USB6008, que atua na aquisição de dados de sensores de temperatura e controla o acionamento das lâmpadas, e o Arduino atuando na recepção dos dados do INA219 e do BH1750 através da comunicação I2C.

O sistema de iluminação artificial é controlado pelo módulo de 4 relés de 5V ao receber comandos do USB6008, que envia sinais para o acionamento ou desativação dos relés, influenciando diretamente na quantidade de luz que incide sobre os módulos.

O LabVIEW serve como a interface do supervisor e plataforma de armazenamento de dados. Ele recebe informações tanto do Arduino quanto do NI USB-6008 através da comunicação serial USB. O supervisor exibe em tempo real os parâmetros medidos.

Figura 4.8: Arquitetura do sistema.



Fonte: Autoria própria, 2024.

---

# Capítulo 5

## Implementação

---

Neste capítulo é descrito o método de implementação do sistema supervisório no software LabVIEW, evidenciando como cada um dos subsistemas foi desenvolvido e como eles são apresentados ao usuário através da tela do supervisório.

### 5.1 Implementação do supervisório no LabVIEW

O LabVIEW é uma plataforma de desenvolvimento baseada em gráficos voltada para aplicações de teste, medição e controle, amplamente utilizada em diversas áreas da engenharia e da ciência, além de ser presente em indústrias. Essa plataforma foi desenvolvida pela National Instruments e permite a criação de sistemas de aquisição de dados, automação de processos e controle industrial, utilizando uma interface de programação gráfica conhecida como G.

Uma das grandes vantagens do LabVIEW é sua capacidade de integrar facilmente hardware de diferentes fabricantes, como placas de aquisição de dados, controladores programáveis e sensores, tornando-se uma ferramenta versátil para monitoramento e controle em tempo real.

No contexto deste projeto, o LabVIEW desempenha um papel crucial como interface supervisória e plataforma de armazenamento de dados, permitindo a aquisição, processamento e visualização dos dados coletados pelo sistema de controle das lâmpadas. Através do LabVIEW, é possível configurar parâmetros de operação, monitorar o estado atual do sistema e armazenar os dados para análise posterior, proporcionando uma visão abrangente do comportamento do sistema ao longo do tempo.

Para elucidar o entendimento acerca dos sistemas implementados no software LabVIEW, eles serão divididos em Subseções para cada um dos sistemas. Contudo, o código completo está anexado ao Apêndice A.2.

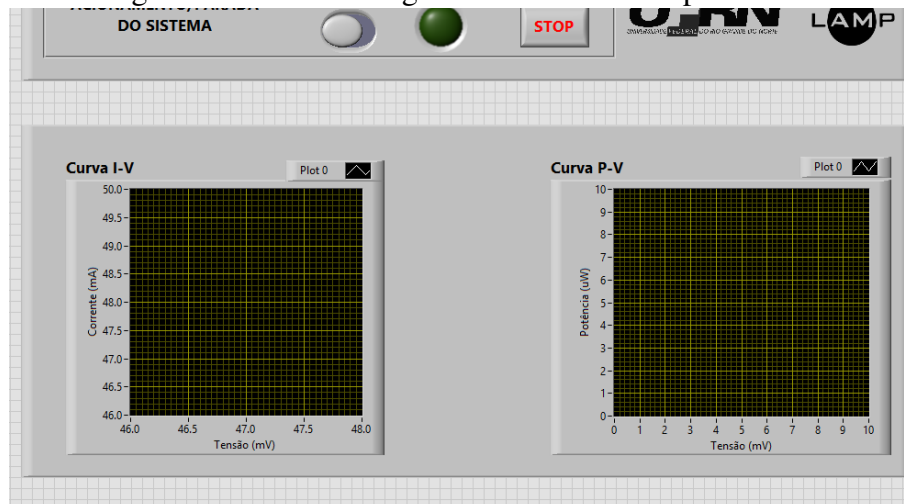
A tela do supervisório foi desenvolvida com o objetivo de apresentar as informações de forma clara ao operador, evidenciando os diferentes sistemas presentes ao separá-los em blocos distribuídos pela tela. Na Figura 5.1 é possível identificar as diferentes seções presentes no sistema, e na Figura 5.2 os gráficos que são plotados.

Figura 5.1: Interface do sistema supervisorio.



Fonte: Autoria própria, 2024.

Figura 5.2: Módulo de gráficos do sistema supervisorio.



Fonte: Autoria própria, 2024.

### 5.1.1 Controle de luminosidade

A intensidade de luminosidade do sistema é definida pela quantidade de lâmpadas ligadas simultaneamente, sendo uma lâmpada equivalente a 25% da luminosidade, e as

quatro lâmpadas o total de 100%. O painel de controle das lâmpadas na tela do supervisor é mostrado na Figura 5.3.

Figura 5.3: Módulo de controle de luminosidade.

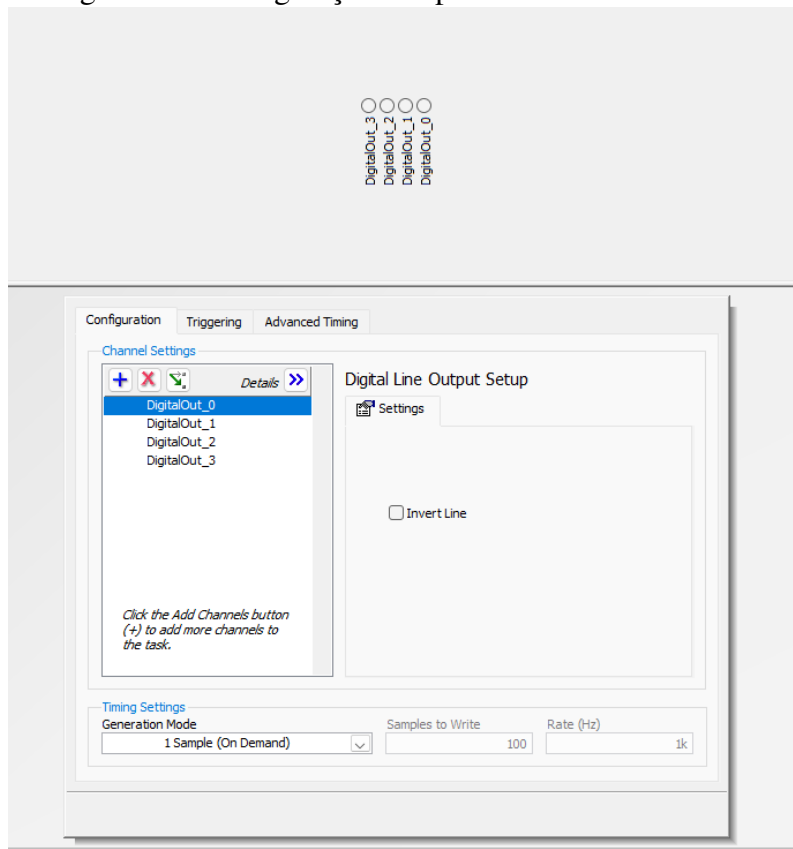


Fonte: Autoria própria, 2024.

Esse controle é feito com o uso do módulo USB6008, configurando quatro pinos digitais como saída no modo de geração “1 Sample (on demand)”, conforme mostrado na Figura 5.4. Essa configuração permite a escrita de um único valor sob demanda, apenas quando solicitado pelo *software*. É comum em aplicações que não requerem aquisição contínua ou amostragem em alta velocidade. O sinal gerado pelo USB6008 é enviado a um módulo com 4 relés (Figura 5.5).

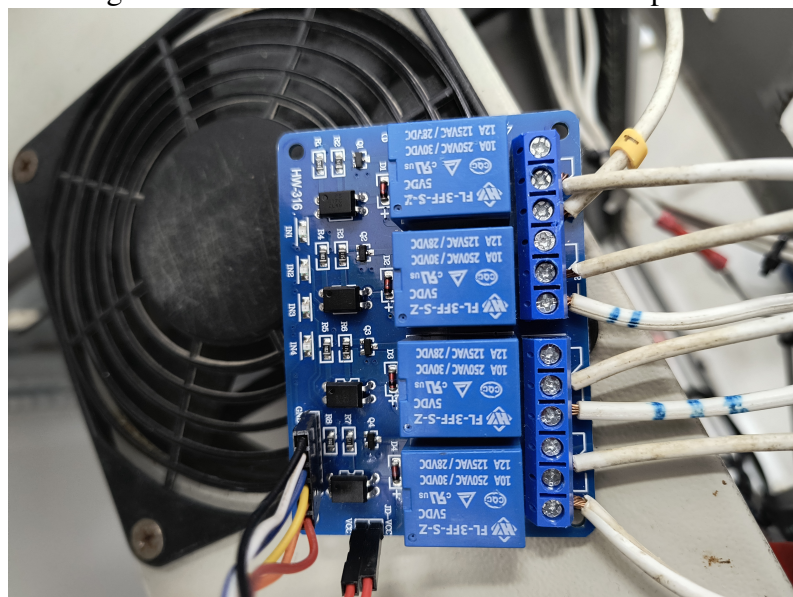
Para a montagem do circuito, a fase foi conectada ao pino comum do relé, e o retorno das lâmpadas ao pino Normalmente Aberto. Assim, o relé começa em Normalmente Fechado, mantendo as lâmpadas desligadas, e ao receber o sinal do DAQ, chaveia para Normalmente Aberto, conforme Figura 5.6.

Figura 5.4: Configuração dos pinos de saída USB6008.



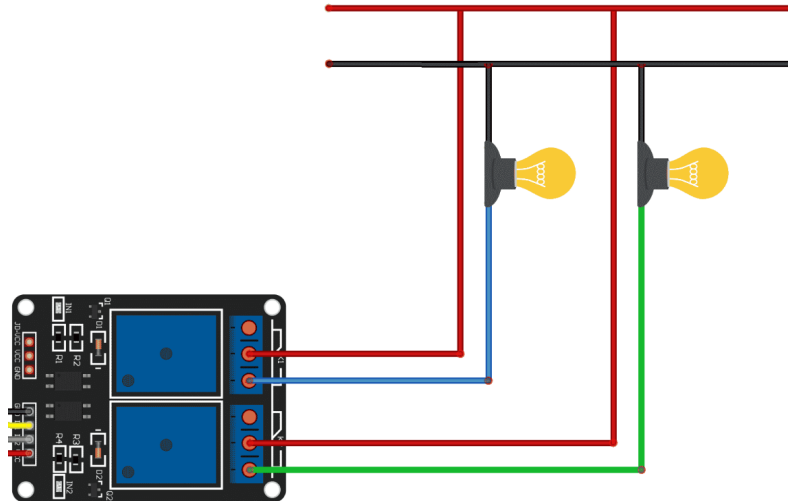
Fonte: Autoria própria, 2024.

Figura 5.5: Módulo 4 relés conectado às lâmpadas.



Fonte: Autoria própria, 2024.

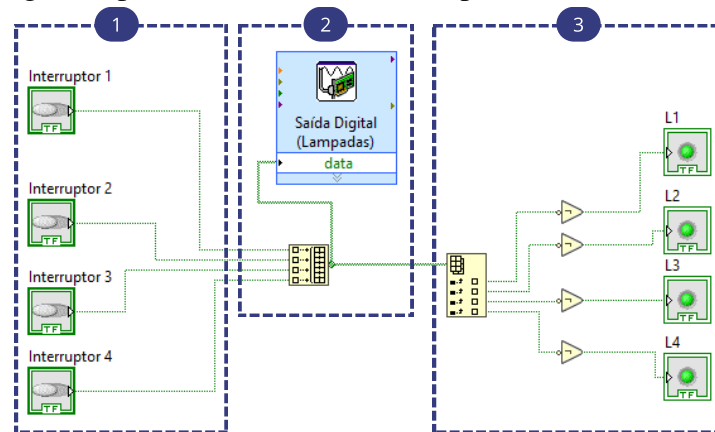
Figura 5.6: Montagem de um circuito de lâmpadas acionadas por relé.  
110/220V



Fonte: Maker, 2025.

A lógica de acionamento das lâmpadas foi dividida em três etapas, conforme Figura 5.7.

Figura 5.7: Lógica implementada em LabVIEW para o acionamento das lâmpadas.



Fonte: Autoria própria, 2024.

1. Os interruptores do supervisor fornecem valores booleanos de *true* ou *false*, conforme seu estado.
2. O LabVIEW cria um vetor com os valores dos interruptores, que o DAQ lê e atribui às saídas analógicas para acionar os relés.
3. O vetor é separado em elementos individuais, que passam por uma porta *NOT* para corrigir a inversão lógica, garantindo que os LEDs acendam corretamente.

### 5.1.2 Monitoramento de tensão, corrente e luminosidade

Para integrar o ambiente LabVIEW com o microcontrolador Arduino, foi necessário utilizar a API de software *Virtual Instrument Software Architecture* (VISA), que fornece uma interface padrão de comunicação através de portas seriais entre o LabVIEW e dispositivos de *hardware*, como Controladores Lógicos Programáveis e instrumentos de medição, como Osciloscópios, Geradores de função, entre outros.

No contexto do projeto, os dados coletados através dos sensores de luminosidade e corrente/tensão são escritos na porta serial do Arduino, e o VISA é utilizado para permitir que ocorra a leitura e operação dos dados obtidos via porta serial. O código carregado no Arduino foi anexado ao apêndice A.1.

A configuração da porta e a visualização dos dados na tela do supervisor pode ser observada na Figura 5.8 a seguir:

Figura 5.8: Configuração de porta e leitura dos dados na tela do supervisor.

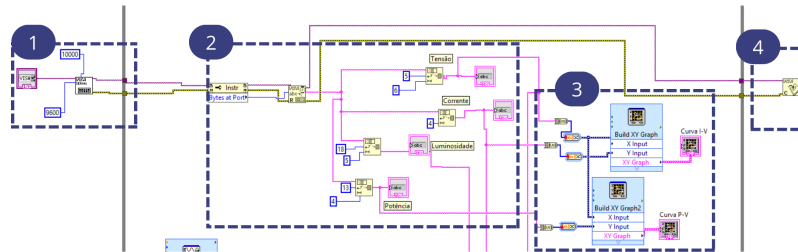


Fonte: Autoria própria, 2024.

Assim como o algoritmo do controle de luminosidade, a lógica implementada para obtenção dos dados de corrente, tensão e luminosidade foi dividida em seções, conforme ilustra a Figura 5.9.

1. Inicialmente é realizada a configuração do *Visa Serial*, informando qual porta o Arduino está conectada, informação do *baud rate* e o *delay* padrão do bloco;
2. Dentro do *loop*, o bloco *Property Nodes* lê as propriedades do objeto (no caso o Arduino) a partir do número de bytes escritos na serial. Como o resultado do *Property Nodes* é um vetor contendo todos os dados coletados pelo Arduino, é necessário separar esses dados utilizando o *String Subset* para dividir o *array* a partir de uma

Figura 5.9: Lógica implementada em LabVIEW para a leitura do Arduino.



Fonte: Autoria própria, 2024.

posição específica e uma quantidade de casas específica. Por exemplo: Para obter a luminosidade está sendo acessado a posição 18 do *array de bytes* e coletando os 5 bytes do 18 em diante;

3. As *strings* particionadas são convertidas em *bytes dinâmicos* para serem exibidos nos gráficos. Os valores também são direcionados para o módulo de armazenamento de dados;
4. Por fim, é encerrada a comunicação do VISA fora do *loop*.

### 5.1.3 Monitoramento de temperaturas

Os valores de temperatura ambiente e temperatura da célula fotovoltaica são obtidos a partir das entradas analógica do USB6008, e exibidas para o usuário conforme mostra a Figura 5.10. Desse modo, foi necessário inicialmente configurar as portas para a coleta dos dados. Para isso, duas portas analógicas foram configuradas como entradas para ler os valores de tensão gerados pelos sensores de temperatura. Além de definir o *range* de tensão na faixa de 0 a 5 volts, também configurou-se o terminal como "*Referenced Single-Ended (RSE)*", tendo em vista que os sensores possuem um mesmo referencial (GND). Por fim, o modo de aquisição foi definido como "*Continuous Samples*", realizando a coleta de forma contínua. Todas as configurações mencionadas podem ser observadas na Figura 5.11.

De modo similar aos códigos anteriores, o algoritmo do monitoramento de temperatura é dividido em três etapas definidas na Figura 5.12.

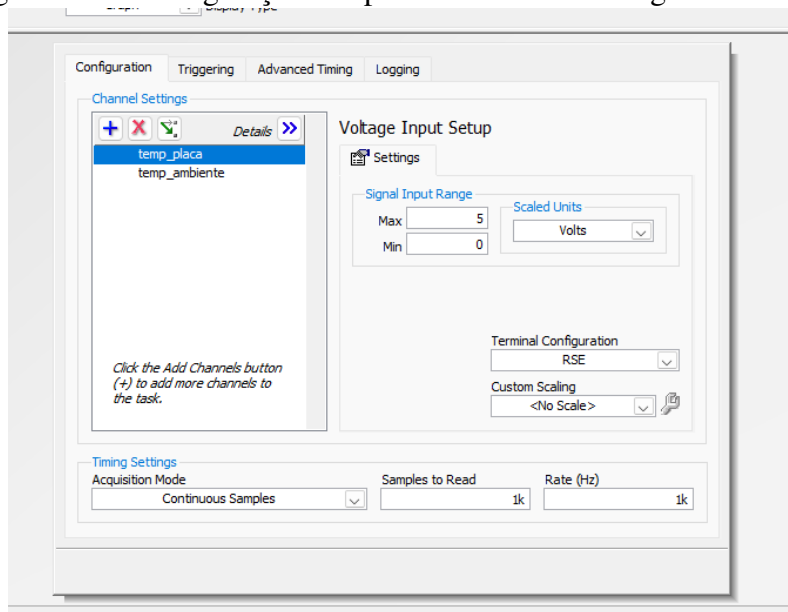
1. No bloco 1, ocorre o recebimento dos dados através das portas de entrada analógica do DAQ, porém há a necessidade de operar e exibir essas temperaturas separadamente. Então, a entrada é transformada em um *Dynamic Data Type (DDT)* com a tag de manipulação de sinal *to DDT*;
2. A entrada é separada em dois valores utilizando o bloco *Index Array*, acessando as posições 0 e 1 do vetor de entrada, que correspondem respectivamente ao valor de

Figura 5.10: Apresentação das temperaturas na tela do supervisor.



Fonte: Autoria própria, 2024.

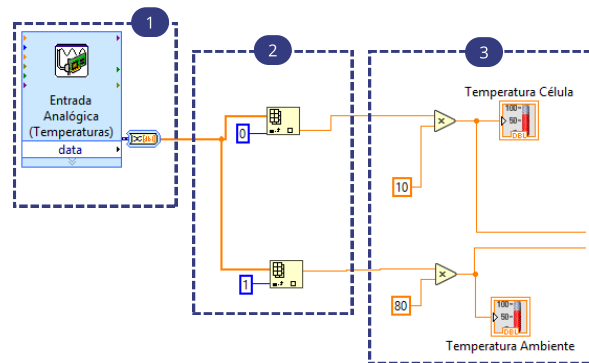
Figura 5.11: Configuração dos pinos de entrada analógica USB6008.



Fonte: Autoria própria, 2024.

- tensão medido pelo sensor presente da Célula fotovoltaica e o sensor que mede a temperatura ambiente;
3. Antes de exibir os valores de temperatura para o usuário, é necessário aplicar um ganho para ajustar a temperatura exibida. O valor de ganho implementado foi obtido de forma empírica.

Figura 5.12: Lógica implementada em LabVIEW para o monitoramento de temperatura.



Fonte: Autoria própria, 2024.

### 5.1.4 Armazenamento de dados

Os dados obtidos através dos sensores são armazenados em arquivos de texto ou planilhas. Para isso, o usuário deve definir o caminho onde o arquivo deve ser armazenado utilizando o campo “Caminho para armazenamento de dados (.txt/.xlsx)” na interface do supervisor apresentado na Figura 5.13, sendo possível sobrescrever um arquivo já existente ou ainda criar um novo arquivo em que ocorrerá o armazenamento dos dados do teste.

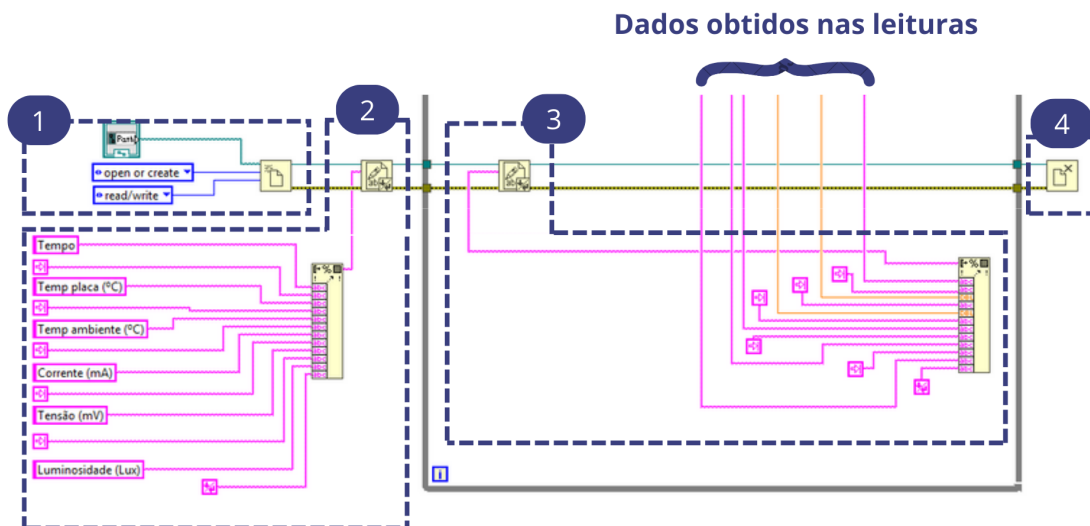
Figura 5.13: Caminho do arquivo na tela do supervisor.



Fonte: Autoria própria, 2024.

A lógica implementada realiza o armazenamento dos dados em *arrays* que contém a hora exata do computador durante a coleta, e os valores de temperatura da célula fotovoltaica e do ambiente, assim como as correntes e tensões geradas pelo painel. A Figura 5.14 apresenta o código em LabVIEW do armazenamento de dados.

Figura 5.14: Código do armazenamento de dados



Fonte: Autoria própria, 2024.

Seguindo as numerações presentes na imagem:

1. O bloco *Open/Create/Read File* é o responsável pela criação do arquivo de armazenamento de dados. Ele recebe como parâmetro o caminho do arquivo fornecido pelo usuário, e as ações a serem realizadas.
2. Na seção 2 ocorre a escrita do cabeçalho do arquivo de dados, ela ocorre fora do *loop* pois será escrita apenas uma vez. Utilizando o bloco *Format into String* é informado todos os parâmetros do cabeçalho, com um espaçamento *tab* entre eles. O bloco *Format into String* fornece a *string* para o bloco *Write to Text File* realizar a escrita da *string*.
3. Dentro do *loop*, similar ao item anterior, é realizada a escrita usando o *Write to Text File* e o *Format into String*, porém escrevendo os dados lidos.
4. Por fim, o bloco *Close File* encerra o arquivo escrito.

---

# Capítulo 6

## Experimentos e Resultados

---

Para validar o funcionamento do sistema supervisorio desenvolvido, foram realizados alguns testes utilizando a bancada mencionada na Seção 2.

### 6.1 Controle de luminosidade

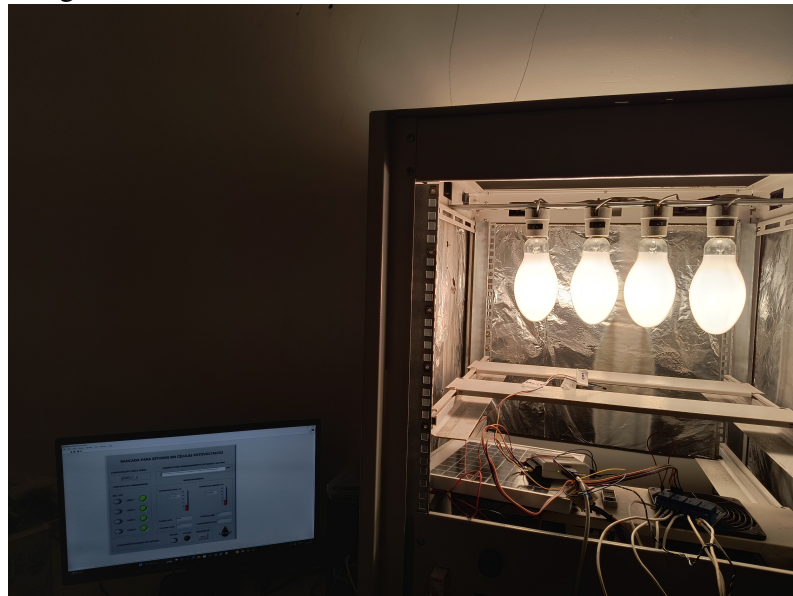
O objetivo do teste é realizar o controle do acionamento das lâmpadas que simulam irradiação solar através do sistema supervisorio. Com o circuito apresentado na seção 5.1.1, o usuário é capaz de controlar o acionamento das lâmpadas através dos comandos gerados pelo USB6008 e enviados para o módulo de quatro relés. As Figuras 6.1 e 6.2 apresentam o monitoramento das lâmpadas acionadas no supervisorio durante testes. É possível observar que o comportamento observado nas lâmpadas presentes na bancada é refletido no sistema supervisorio.

Figura 6.1: Sistema acionado com apenas 50% de luminosidade total.



Fonte: Aatoria própria, 2024.

Figura 6.2: Sistema acionado com 100% de luminosidade.



Fonte: Autoria própria, 2024.

## 6.2 Monitoramento de tensão, corrente e luminosidade

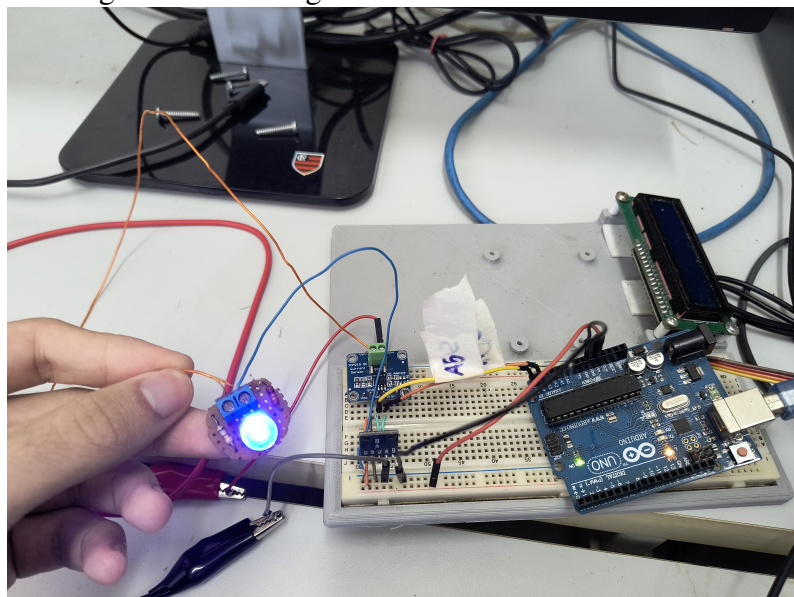
Para validar a supervisão de corrente e tensão, foi montado um circuito auxiliar utilizando o INA219 aferindo os valores de um LED (ver Figura 6.3), pois a bancada de estudos não estava operacional. Tendo em vista que o objetivo do trabalho é validar o funcionamento do sistema supervisor, optou-se por validá-lo monitorando um LED como carga.

Para a obtenção da luminosidade, o BH1750 foi posicionado próximo às lâmpadas, para ser capaz de responder a variações de luminosidade controladas pelo acionamento da iluminação artificial. A resposta do sensor de luminosidade observada na Figura 6.4 é um valor na unidade Lux, que é a quantidade de luz que atinge uma área de 1 metro quadrado ( $m^2$ ). Uma forma comum de interpretar a unidade Lux é:

- 0 lux: Total escuridão;
- 10 lux: Luz de uma vela vista de perto;
- 100 lux: Iluminação de ambiente interno moderado (uma lâmpada de 100W a cerca de 1 metro de distância);
- 1000 lux: Iluminação mais intensa, como em uma sala bem iluminada ou sob luz fluorescente;
- 10.000 lux ou mais: Luz intensa como a luz solar direta em um dia claro.

Embora o Arduino esteja realizando a comunicação com o sistema de maneira correta, há um *delay* de cerca de 10 segundos entre o que é escrito na serial e o que o LabVIEW lê. Acredita-se que esse *delay* é causado pela configuração padrão do VISA no LabVIEW.

Figura 6.3: Montagem do circuito teste com o LED.



Fonte: Autoria própria, 2024.

Figura 6.4: Teste de monitoramento do INA219 e BH1750.



Fonte: Autoria própria, 2024.

### 6.3 Monitoramento de temperaturas

Para o monitoramento das temperaturas da célula fotovoltaica e do ambiente, os sensores LM35 e LM335 foram posicionados na bancada de modo que um deles estivesse

mais próximo às lâmpadas e o outro mais distante, como aconteceria no funcionamento convencional da bancada. Os valores monitorados são apresentados na interface e podem ser observados na Figura 6.5

Figura 6.5: Monitoramento de temperaturas com as 4 lâmpadas ligadas.



Fonte: Autoria própria, 2024.

O usuário é capaz de monitorar em tempo real os valores medidos das temperaturas pelo supervisor e, em caso de superaquecimento ou qualquer comportamento inesperado, o operador é capaz de tomar uma rápida decisão de interromper o teste e preservar a integridade do sistema.

## 6.4 Armazenamento de dados

Os dados coletados durante o teste são armazenados em planilhas .XLSX ou .TXT, como mostra a Figura 6.6. De posse desses dados, é possível gerar gráficos e realizar análises dos testes, comparando-os entre si, além de possibilitar um histórico de desempenho do sistema sob diferentes condições. Uma outra utilidade dos dados é a detecção de anomalias do sistema que podem passar despercebidas se ocorrerem rapidamente.

Figura 6.6: Foto dos dados salvos formato .xlsx.

|     | A        | B               | C                  | D             | E           | F                  |
|-----|----------|-----------------|--------------------|---------------|-------------|--------------------|
| 1   | Tempo    | Temp placa (°C) | Temp ambiente (°C) | Corrente (mA) | Tensão (mV) | Luminosidade (Lux) |
| 110 | 16:43:23 | 25.464.060      | 37.955.018         | 0.60          | 2736.0      | 6                  |
| 111 | 16:43:24 | 25.464.060      | 37.955.018         | 0.10          | 2732.0      | 6                  |
| 112 | 16:43:25 | 25.566.126      | 38.771.568         | 0.60          | 2736.0      | 343                |
| 113 | 16:43:26 | 25.464.060      | 38.771.568         | 0.30          | 2732.0      | 1131               |
| 114 | 16:43:27 | 25.464.060      | 39.588.117         | 0.30          | 2732.0      | 1758               |
| 115 | 16:43:28 | 25.464.060      | 39.588.117         | 0.40          | 2728.0      | 2098               |
| 116 | 16:43:29 | 25.566.126      | 39.588.117         | 0.50          | 2732.0      | 2143               |
| 117 | 16:43:30 | 25.566.126      | 40.404.666         | 0.50          | 2732.0      | 2175               |
| 118 | 16:43:31 | 25.566.126      | 40.404.666         | 0.40          | 2732.0      | 2527               |
| 119 | 16:43:32 | 25.566.126      | 41.221.215         | 0.10          | 2732.0      | 2589               |
| 120 | 16:43:33 | 25.566.126      | 41.221.215         | 0.40          | 2732.0      | 2750               |
| 121 | 16:43:34 | 25.566.126      | 41.221.215         | 0.30          | 2732.0      | 2839               |
| 122 | 16:43:35 | 25.566.126      | 42.037.764         | 0.20          | 2736.0      | 2967               |
| 123 | 16:43:36 | 25.668.192      | 42.037.764         | 0.30          | 2732.0      | 3029               |
| 124 | 16:43:37 | 25.566.126      | 42.037.764         | 0.40          | 2736.0      | 3062               |
| 125 | 16:43:38 | 25.668.192      | 42.037.764         | 0.50          | 2736.0      | 3064               |
| 126 | 16:43:39 | 25.668.192      | 42.854.313         | 0.20          | 2732.0      | 3068               |
| 127 | 16:43:40 | 25.668.192      | 42.037.764         | 0.20          | 2732.0      | 3058               |
| 128 | 16:43:41 | 25.668.192      | 42.037.764         | 0.60          | 2744.0      | 3065               |
| 129 | 16:43:42 | 25.668.192      | 42.037.764         | 0.60          | 2772.0      | 3074               |
| 130 | 16:43:43 | 25.668.192      | 42.037.764         | 0.60          | 2804.0      | 3105               |

Fonte: Autoria própria, 2024.

---

# Capítulo 7

## Conclusão

---

O desenvolvimento do sistema supervisório proposto para o monitoramento e controle de uma bancada de estudos em energia fotovoltaica mostrou resultados satisfatórios, possibilitando a aquisição e análise em tempo real de parâmetros críticos, como tensão, corrente, luminosidade, e temperaturas da placa fotovoltaica e do ambiente. A integração do sistema com a plataforma LabVIEW proporcionou uma interface amigável para os usuários, facilitando a supervisão e o armazenamento dos dados.

Embora funcional, o sistema apresenta algumas limitações, pois a velocidade de transferência de dados e taxa de atualização entre o Arduino e o LabVIEW apresentam um *delay* significativo, podendo prejudicar execução de testes. A comunicação pode ser otimizada para suportar maiores volumes de dados ou velocidades de transmissão, especialmente em testes de longa duração ou sob condições variáveis. Além disso, implementação da interface gráfica para análise das curvas não está 100% funcional.

Os resultados obtidos evidenciam a capacidade do uso do supervisório desenvolvido para aplicações práticas em estudos de energia fotovoltaica, destacando-se como uma ferramenta valiosa para o monitoramento e a análise de desempenho capaz de oferecer aos pesquisadores um maior controle acerca dos processos que ocorrem durante os testes da bancada. Entretanto, há espaço para aprimoramentos que possam aumentar ainda mais a eficiência, precisão e funcionalidade do sistema, oferecendo maior confiabilidade para os usuários.

### 7.1 Trabalhos futuros

Desse modo, algumas sugestões de novos trabalhos em cima dessa temática podem ser feitas para os projetos do supervisório e da bancada de estudo. Entre elas, destacam-se principalmente a implementação da simulação de passagem de nuvens, que pode ser feita utilizando algum material translúcido que gere um sombreamento e seja capaz de reter parte da luminosidade gerada pelas lâmpadas.

O desenvolvimento de um algoritmo capaz de rastrear o MPPT e ajustá-lo automaticamente para que a célula fotovoltaica permaneça em seu ponto de operação o máximo possível pode ser uma grande melhoria para o sistema. Uma outra sugestão de experimento seria o controle de direção ou inclinação do painel.

Essas funcionalidades implementadas podem ser controladas através do supervisório,

como controle de parâmetros de rastreamento do MPPT e variável de inclinação da placa. O sistema de supervisão poderia, também, apresentar uma melhora no tratamento de erros, sendo capaz de identificar um erro grave como aumento desproporcional de corrente no circuito, e desligar o sistema automaticamente como medida de segurança.

---

## Referências Bibliográficas

---

- Abdelaziz, Almoataz Y & Y Almoataz (2020), *Modern maximum power point tracking techniques for photovoltaic energy systems*, Springer.
- Amazon (n.d.), ‘Sensor de temperatura lm35 para arduino e raspberry’, <https://www.amazon.com.br/Sensor-Temperatura-Lm35-Arduino-Raspberry/dp/B07Q2F715B>. Acessado em: 19 jan. 2025.
- Arduino e Cia (n.d.), ‘Sensor de corrente dc ina219’, <https://www.arduinoecia.com.br/product/sensor-de-corrente-dc-ina219/>. Acessado em: 19 jan. 2025.
- COELHO, Marcelo S (2010), ‘Apostila de sistemas supervisórios’, *Curso superior de tecnologia em automação e controle de processos industriais contínuos. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo Campus Cubatão*.
- DeD Componentes Eletrônicos (n.d.), ‘Arduino uno r3 original da itália’, <https://www.dedcomponentes.com.br/produto/arduino-uno-r3-original-da-italia/>. Acessado em: 19 jan. 2025.
- Erriez (n.d.), ‘Erriez bh1750 digital light sensor library’, <https://github.com/Erriez/ErriezBH1750>. Acessado em: 19 jan. 2025.
- Gates, Luis F. (2000), *Telecomunicações Baseadas na Arquitetura BALSE*, Tese de doutorado, UFSCG, Matosinhos, SP.
- Gates, Luis F., Rossano P. Pinto, Eliane G. Solimões & Katianna Silva (2001), *Mobile agents in telecommunication services*, em J. C.A. Loureiro, ed., ‘Proceedings of LACOXS’05’, Sociedade Brasileira de Ciências Físicas, Ouro Preto, MG, pp. 50–61.
- Governo do Brasil (2021), ‘Energia renovável chega a quase 50% da matriz elétrica brasileira’. Acessado em: 2025.  
**URL:** <https://www.gov.br/pt-br/noticias/energia-minerais-e-combustiveis/2021/08/energia-renovavel-chega-a-quase-50-da-matriz-eletrica-brasileira-1>
- Industries, Adafruit (n.d.), ‘Adafruit bh1750 light sensor’, <https://www.adafruit.com/product/469>. Acessado em: 19 jan. 2025.

- INPE (2017), *Atlas Brasileiro de Energia Solar*, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, Brasil. Acesso em: 14 jan. 2025.  
**URL:** <http://www.dsr.inpe.br/sbsr2017/files/p0071.pdf>
- Instruments, National (n.d.a), 'Usb-6008 specifications', <https://www.digikey.com.br/pt/products/detail/ni/779051-01/12817874>. Acessado em: 19 jan. 2025.
- Instruments, National (n.d.b), 'Usb-6008/6009 pinout and signal descriptions'. Acessado: 2024.  
**URL:** <https://www.ni.com/docs/en-US/bundle/usb-6008-6009-feature/page/pinout-signal-descrip.html>
- Instruments, Texas (n.d.a), 'Lm335 data sheet', <https://www.ti.com/lit/ds/symlink/lm335.pdf>. Acessado em: 19 jan. 2025.
- Instruments, Texas (n.d.b), 'Lm35 precision centigrade temperature sensors', <https://www.ti.com/product/LM35>. Acessado em: 19 jan. 2025.
- IONA (1999), *Orbix Programming Reference*, IONA Technologies PLC, Dublin, Ireland.
- MakerHero (2025), 'Controle de módulo relé com arduino'. Acessado em: 2025.  
**URL:** <https://www.makerhero.com/blog/controle-modulo-rele-arduino/>
- Mankar, Jayant, Chaitali Darode, Komal Trivedi, Madhura Kanoje & Prachi Shahare (2014), 'Review of i2c protocol', *International Journal of Research in Advent Technology* 2(1).
- Marcondes, Danilo (2005), *Automação Industrial: Controle do Movimento e Processos Contínuos*, Editora Érica, São Paulo.
- Marini, José Adriano & Luiz Antonio Rossi (2002), 'Projeto de sistemas fotovoltaicos para oferta de energia elétrica a comunidades rurais', *Proceedings of the 4th Encontro de Energia no Meio Rural*.
- Meireles, Flamingo (2003), NOVET's homepage, Página na internet, The Academic Computer Society, Caicó, RN.  
**URL:** <http://www.aligatesta.com.su/>
- Mittelbach, Frank & Michel Goossens (2004), *The L<sup>A</sup>T<sub>E</sub>X Companion*, 2ª edição, Addison-Wesley, San Francisco, EUA.
- National Instruments (2018), *NI USB-6008 Specifications*. Accessed: 2025-01-18.  
**URL:** <https://www.ni.com/docs/en-US/bundle/usb-6008-specs/page/specs.html>
- OMG (2000), Audio/video streams, version 1.0, OMG Specification formal/2000-01-03, Object Management Group, Paris.  
**URL:** <http://www.omg.org/>

- Pedrycz, Witold & Fernando A. C. Gomide (1998), *An Introduction to Fuzzy Sets: Analysis and Design (Complex Adaptive Systems)*, MIT Press, Boston.
- Pinto, Alexandre S., Luis F. Gates & Katianna Silva (2000), A BALSE-based environment for mobile applications, *em* J. N.de Souza & R.Boutaba, eds., 'Managing QoS in Multimedia Networks', Kluwer Academic Publishers, New York, pp. 163–175.
- Pirmez, Luci, Luis Fernando Carmo & Raimundo José Macêdo, eds. (2002), *XX Simpósio Brasileiro de Redes de Computadores*, Vol. 1, Núcleo de Computação Eletrônica da UFRJ, Sociedade Brasileira de Computação, Rio de Janeiro.
- Relay, Songle (n.d.), 'Srd-05vdc-sl-c datasheet', <https://www.alldatasheet.com/datasheet-pdf/view/1131944/SONGLERELAY/SRD05VDCSLC.html>. Acessado em: 19 jan. 2025.
- Salazar, Andres O, Diego A de M Fonseca, Caio ML Ribeiro, Vinicius M Xavier, Gabriela MS Lopes, Kalynne G da S Maciel et al. (2023), 'Study of the relationship between temperature and efficiency in photovoltaic systems', *RE&PQJ* **21**(5).
- Salazar, Andres Ortiz & RM Stephan (1993), 'A bearingless method for induction machines', *IEEE Transactions on Magnetics* **29**(6), 2965–2967.
- Santos, Euripedes P. & Fernando J. Von-Zuben (2000), *Recurrent Neural Networks: Design and Applications*, Vol. 13 de *International Series on Computational Intelligence*, CRC Press, Madrid, capítulo Efficient Second-Order Learning Algorithms for Discrete-Time Recurrent Neural Networks, pp. 47–75.
- Santos-Filho, José C. S. (2003), Combinação por ganho igual em ambiente com desvanecimento arbitrário, Dissertação de mestrado, Faculdade de Engenharia Elétrica e de Computação, UNICAMP, Campinas, SP.
- Sichman, Jaime S. (2002), Sobre agentes móveis e fixos. Comunicação pessoal por e-mail.
- Solar, Canal (n.d.), 'Inversores com múltiplos mppt'. Acessado: 2024.  
**URL:** <https://canalsolar.com.br/inversores-com-multiplos-mppt/>
- Solimões, Eliane G., Antonio T. Mircolles, Jamerson L. Pereira, Bruno G. Americano, Katianna Silva, Marcelo Borges & Mauricio F. Magalhães (2003), 'A virtual laboratory for autonomous airplanes experiments', *IEEE Transactions on Education* **46**(1), 37–42.
- Souza, Rodrigo B, Adelardo AD Medeiros, João MA Nascimento, André L Maitelli & Heitor P Gomes (2006), Sisal, um sistema supervisor para elevação artificial de petróleo, *em* 'Rio Oil & Gas Expo and Conference', pp. 11–14.
- STA Eletrônica (2025), 'Características principais do arduino uno'. Acessado em: 18 jan. 2025.

**URL:** *<https://www.sta-eletronica.com.br/artigos/arduinos/caracteristicas-principais-do-arduino-uno>*

Wantronics (n.d.), 'Comunicação i2c: O que é e como funciona', <https://wantronics.com.br/2024/01/01/comunicacao-i2c/>. Acessado em: 19 jan. 2025.

---

# Apêndice A

## Informações adicionais

---

### A.1 Algoritmo Arduino

```
#include <Wire.h>
#include <Adafruit_INA219.h>
#include <BH1750.h>

// Instanciação dos sensores
Adafruit_INA219 ina219;
BH1750 lightMeter;

void setup() {
  // Inicialização da comunicação serial
  Serial.begin(9600);

  // Inicialização do sensor INA219
  if (!ina219.begin()) {
    Serial.println("Falha ao inicializar o INA219. Verifique as conexões!");
    while (1);
  }

  // Inicialização do sensor BH1750
  lightMeter.begin();
}

void loop() {
  // Leitura de valores do INA219
  float shuntvoltage = ina219.getShuntVoltage_mV();
  float busvoltage = ina219.getBusVoltage_V();
  float current_mA = ina219.getCurrent_mA();
  float voltage = busvoltage + (shuntvoltage / 1000);

  // Leitura de valores do BH1750
```

```

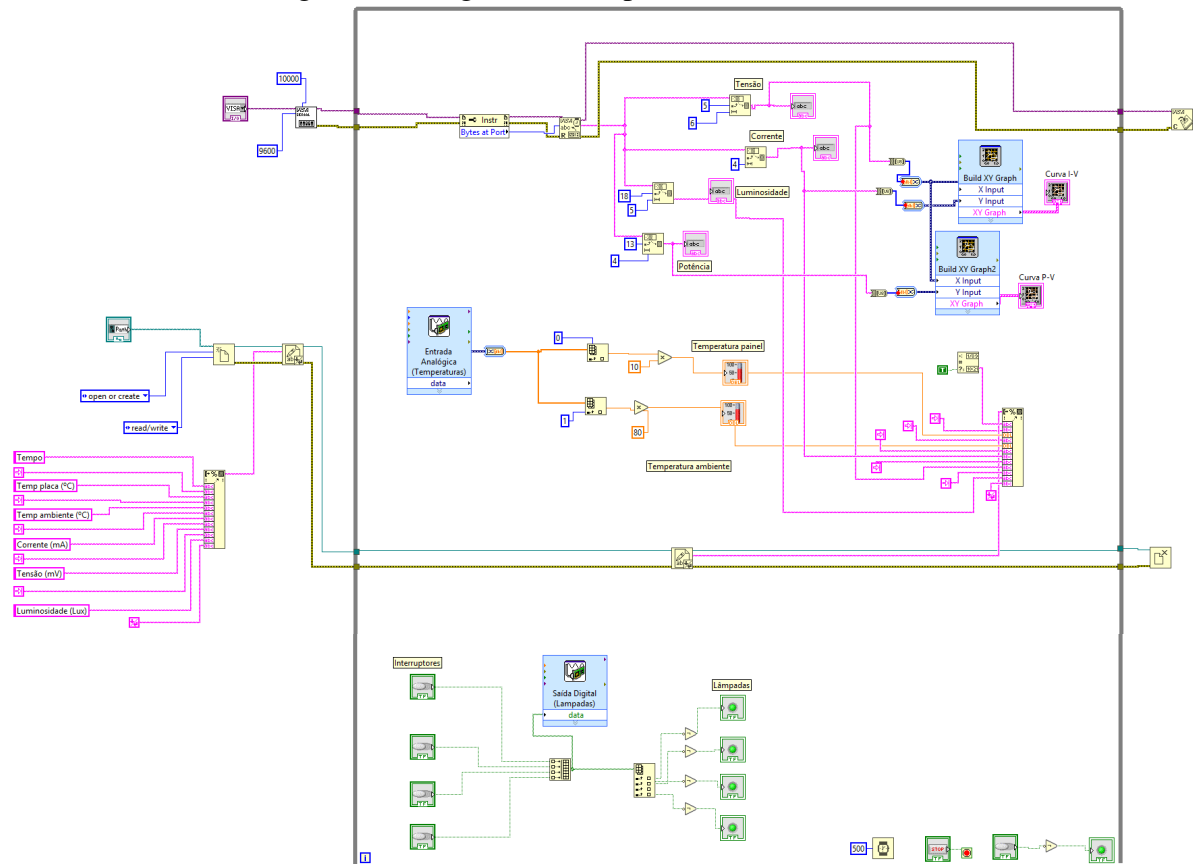
uint16_t lux = lightMeter.readLightLevel();

// Exibição dos valores na serial no formato especificado
Serial.print(current_mA);
Serial.print(";");
Serial.print(voltage * 1000); // Convertendo para mV
Serial.print(";");
Serial.print(voltage * current_mA);
Serial.print(";");
Serial.println(lux);

delay(500); // Aguarda 500ms antes da próxima leitura
}
    
```

## A.2 Código completo LabVIEW

Figura A.1: Algoritmo completo em LabVIEW.



Fonte: Autoria própria, 2024.