



**UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE ENGENHARIA**

**DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA
CURSO DE ENGENHARIA ELECTRÓNICA**

**Proposta de um sistema de medição e
controle do consumo de energia eléctrica**

Trabalho de Licenciatura

Gustavo Zacarias Guambe

**Supervisor: Eng.^a Ivone Cipriano
(UEM, Faculdade de Engenharia, Departamento de Engenharia Electrotécnica)**

Maputo, Outubro 2025



**UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE ENGENHARIA**

**DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA
CURSO DE ENGENHARIA ELECTRÓNICA**

**Proposta de um sistema de medição e controle do
consumo de energia eléctrica**

Trabalho de Licenciatura

Gustavo Zacarias Guambe

**Supervisor: Eng.^a Ivone Cipriano
(UEM, Faculdade de Engenharia, Departamento de Engenharia Electrotécnica)**

Maputo, Outubro 2025

GUSTAVO ZACARIAS GUAMBE

Proposta de um sistema de medição e controle do consumo de energia eléctrica

Monografia apresentada ao Departamento de Engenharia Electrotécnica da Faculdade de Engenharia da Universidade Eduardo Mondlane – como requisito parcial para obtenção do grau de Licenciatura em Engenharia Electrónica.

Supervisor: Eng.^a Ivone Cipriano
(UEM, Faculdade de Engenharia, Departamento de Engenharia Electrotécnica)

Maputo, Outubro 2025

DECLARAÇÃO DE HONRA

Declaro sobre palavra de honra que este trabalho foi feito totalmente por minha autoria.

EPÍGRAFE

Não se pode controlar o que não se pode medir.

Peter Drucker

DEDICATÓRIA

Sem a direção dada por Deus, a conclusão deste trabalho não seria possível. Por causa disso, dedico este trabalho de licenciatura a Ele. Com muita gratidão no coração.

AGRADECIMENTOS

Durante a elaboração deste trabalho assim como ao longo do curso, muitas pessoas de forma generosa dedicaram seu tempo e esforço para tornar esta conquista viável.

Enfrentei diversos desafios ao longo do caminho e posso afirmar com certeza que não teria conseguido superá-los sem o apoio fundamental de minha família, amigos e colegas. Primeiramente, agradeço a Deus por abençoar-me com a vida, saúde e com a possibilidade de estar concluindo este curso.

Agradeço profundamente à minha família pelo apoio constante ao longo de todo o curso e em todos os momentos da minha vida. A família é, sem dúvida, o maior tesouro que podemos ter.

Agradeço aos meus amigos e colegas de turma que me acompanharam ao longo do curso, por todo o auxílio e compreensão proporcionadas. Que possamos manter esta amizade por muito tempo.

Quero agradecer à minha supervisora, a Eng.^a Ivone Cipriano, por um lado na posição de supervisora, que foi desde o início do trabalho uma boa orientadora, e por outro, como docente, contribuindo significativamente para minha formação.

Sou grato a todos que estiveram ao meu lado e compartilharam um pouco de suas vidas comigo. A todos, deixo um sincero e enorme agradecimento!

RESUMO

Proposta de um sistema de medição e controle do consumo de energia eléctrica

A presente monografia tem como objectivo desenvolver um sistema de medição e controle do consumo de energia eléctrica de duas cargas monofásicas, correspondentes aos sistemas de iluminação e de condicionamento ambiental, instalados em cômodos distintos de um ambiente. O sistema é composto por dois subsistemas principais: medição e controle. Analisou-se as formas de medição e controle do consumo de energia eléctrica, tendo-se identificado os seguintes componentes para a implementação do subsistema de medição: sensores de corrente e tensão, microcontroladores Arduino Uno e ESP32, e um display LCD para visualização local dos dados. Esses sensores realizam a leitura dos valores eléctricos, permitindo o cálculo da potência instantânea e do consumo acumulado de energia. As informações obtidas são disponibilizadas localmente e também remotamente através de uma interface web, que oferece ainda o envio de notificações ao utilizador sobre o consumo energético. Para o subsistema de controle, foram utilizados relés e os sensores específicos. No cômodo com o sistema de iluminação, foi integrado um sensor de presença que permite o acionamento automático da luz apenas na presença de pessoas, desligando-a na ausência. Já no cômodo com o sistema de climatização, foi utilizado um sensor de temperatura para o controle automático do condicionador de ar, em função das condições térmicas do ambiente. A lógica de funcionamento do sistema foi desenvolvida utilizando a linguagem de programação C++, enquanto a interface web foi construída com HTML, proporcionando uma interação simples e eficaz.

Palavras Chaves: Medição de energia eléctrica, Controle de cargas, Linguagem C, Programação Web, Eficiência energética.

ABSTRACT

Proposal for a measurement and control system of electric energy consumption

This monograph aims to develop a system for measuring and controlling the electrical energy consumption of two single-phase loads, corresponding to the lighting and air conditioning systems installed in separate rooms of an environment. The system is composed of two main subsystems: measurement and control. The methods for measuring and controlling energy consumption were analyzed, and the following components were identified for the implementation of the measurement subsystem: current and voltage sensors, Arduino Uno and ESP32 microcontrollers, and an LCD display for local data visualization. These sensors read the electrical values, enabling the calculation of instantaneous power and accumulated energy consumption. The obtained information is made available both locally and remotely through a web interface, which also allows for sending notifications to the user about energy consumption. For the control subsystem, relays and specific sensors were used. In the room with the lighting system, a presence sensor was integrated, allowing the lights to be automatically switched on when motion is detected and turned off in the absence of people. In the room with the air conditioning system, a temperature sensor was used to automatically control the air conditioner based on the thermal conditions of the environment. The system's logic was developed using the C++ programming language, while the web interface was built using HTML, providing a simple and effective user interaction.

Keywords: Electric energy measurement, Load control, C programming language, Web programming, Energy efficiency.

Índice de Conteúdo

Índice	vii
Lista de Figuras	xi
Lista de Acrónimos	xv
1 Contextualização do tema	1
1.1 Introdução	1
1.2 Formulação do problema	2
1.3 Justificativa	3
1.4 Objectivos	3
1.4.1 Objectivo Geral	3
1.4.2 Objectivos Específicos	3
1.5 Metodologia de investigação	4
1.5.1 Classificação da Pesquisa	4
1.5.1.1 Quanto à natureza	4
1.5.1.2 Quanto à abordagem	4
1.5.1.3 Quanto aos objectivos	4
1.5.1.4 Quanto à técnica de pesquisa	4
1.5.2 Ferramentas Utilizadas	5
1.6 Estrutura do trabalho	5
2 Revisão Teórica	6
2.1 Energia Eléctrica	6
2.1.1 Análise da energia eléctrica em <i>Corrente Alternada</i> (CA)	6
2.1.1.1 Características da CA	7
2.1.1.2 POTÊNCIAS INSTANTÂNEA, ACTIVA, REACTIVA E APARENTE	9

2.1.2	Medidores de Energia Eléctrica	12
2.1.2.1	Medidores Electromecánicos	13
2.1.2.2	Medidores Electrónicos	13
2.1.3	Classificação de cargas eléctricas	14
2.1.4	Consumo e tarifas de electricidade	16
2.2	Sistemas de controle	18
2.2.1	Configurações dos sistemas de controle	19
2.2.1.1	Sistemas de controle de malha aberta	19
2.2.1.2	Sistemas de controle de malha fechada	19
2.2.2	Sistema de controle do consumo de energia	20
2.2.2.1	Componentes de um sistema de controle de energia	21
2.3	Sensores	21
2.3.1	Sensor de corrente invasivo	21
2.3.2	Sensor de Tensão	23
2.3.3	Sensor de presença	25
2.3.4	Sensor de temperatura	27
2.4	Microcontroladores	29
2.4.1	Plataforma Arduino	29
2.4.2	ESP32	31
2.5	Actuadores	33
2.5.1	Módulo Relé	33
2.6	Programação	34
2.6.1	Tipos de linguagens de programação	35
2.6.2	Linguagem C++	35
2.7	Web	36
2.7.1	Programação web	37
2.7.1.1	HTML (HyperText Markup Language)	37
2.7.1.2	CSS (Cascading Style Sheets)	37
2.7.1.3	JavaScript	37
3	Desenho & Implementação	38
3.1	Dimensionamento do sistema	39
3.1.1	Subsistema de medição do consumo de energia	39
3.1.1.1	Esquema eléctrico	39

3.1.1.2	Componentes	40
3.1.2	Subsistema de controle do consumo de energia	41
3.1.2.1	Esquema eléctrico	41
3.1.2.2	Componentes	42
3.1.3	Linguagem de programação usada	43
3.1.4	Página web	43
3.2	Implementação do sistema	43
3.2.1	Descrição do sistema proposto	43
3.2.1.1	Diagrama de blocos	44
3.2.2	Subsistema de medição do consumo de energia	44
3.2.3	Subsistema de controle do consumo de energia	45
3.2.4	Código fonte	46
3.2.4.1	Código fonte do subsistema de medição do consumo de energia	46
3.2.4.2	Código fonte do subsistema de controle do consumo de energia	48
3.2.5	Página web	50
3.2.5.1	Estrutura Geral da Página	50
3.2.5.2	Cabeçalho (<head>)	51
3.2.5.3	Corpo da Página (<body>)	51
3.2.5.4	Notificações	52
3.2.5.5	Comunicação com o ESP32 (WebSocket)	52
3.2.5.6	Lógica de Cálculo do Consumo	53
3.2.5.7	Alertas e Comportamento do Sistema	53
4	Testes e Resultados	54
4.1	Protótipo do sistema	55
4.2	Testes no subsistema de medição do consumo de energia	55
4.3	Testes no subsistema de controle do consumo de energia	57
4.3.1	Testes no cómodo da carga de ventilação	57
4.3.2	Testes no cómodo da carga de iluminação	59
4.4	Testes na página web	61
4.4.1	Notificações e entrada de dados	62

5 Conclusão	65
5.1 Recomendação para trabalhos Futuros	66
Referências Bibliográficas	67
Anexos	70
1 Código 1	1
1.1 Código do medidor	1
2 Código 2	5
2.1 Código da página web	5
3 Código 3	11
3.1 Código do sistema de controle	11

Lista de Figuras

2.1	Projecção de um vector. Fonte: [(Alves, 1999)]	7
2.2	Representação gráfica de período. Fonte: [(Alves, 1999)]	8
2.3	Representação gráfica de amplitude máxima. Fonte: [(Alves, 1999)]	9
2.4	Gráfico de potência instantânea para diferentes circuitos. Fonte: [(Alves, 1999)]	10
2.5	Gráfico de potência instantânea para indutância pura. Fonte: [(Alves, 1999)]	11
2.6	Triângulo de potências. Fonte: [(Alves, 1999)]	12
2.7	Medidor Electromecânico Fonte: https://pt.wikipedia.org/w/index.php?title=Medidor_de_energia_el%C3%A9tricaaction=info	13
2.8	Medidor Electrónico. Fonte: https://pt.wikipedia.org/w/index.php?title=Medidor_de_energia_el%C3%A9tricaaction=info	14
2.9	Curva típica de carga residencial. Fonte:[(Vasconcelos, 2017)]	16
2.10	Curva típica de carga comercial. Fonte:[(Vasconcelos, 2017)]	16
2.11	Descrição simplificada de um sistema de controle. Fonte:[(Nise, 2013)]	18
2.12	Diagrama de blocos de um sistema de controle em malha aberta. Fonte:[(Nise, 2013)]	19
2.13	Diagrama de blocos de um sistema de controle em malha fechada. Fonte:[(Nise, 2013)]	20
2.14	Sensor de corrente ACS712. Fonte: https://www.usinainfo.com.br/blog/projeto-medindo-corrente-com-o-sensor-ac712-e-o-arduino/	22
2.15	Curva característica do sensor de corrente ACS712. Fonte: https://www.allegromicro.com//datasheets/acs712-datasheet	23
2.16	Sensor de Tensão AC ZMPT101B. Fonte: https://www.casadarobotica.com/sensores-modulos/sensores/tensao/sensor-de-tensao-ac-0-a-250v-zmpt101b	24
2.17	Esquema eléctrico do sensor de tensão AC ZMPT101B. Fonte: https://www.fernandok.com/2022/04/medidor-de-tensao-e-corrente-ac-com.html	24

2.18 Sensor PIR. Fonte: https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf	25
2.19 Processo de funcionamento do sensor pir. Fonte: https://forum.automation.com/t/motion-detection-using-pir-passive-infrared-sensor/5485	26
2.20 Pinagem do sensor pir. Fonte: https://www.bosontreinamentos.com.br/electronica/como-funciona-um-sensor-de-movimento-pir-passive-infrared/	27
2.21 Sensor de temperatura. Fonte: https://www.alldatasheet.com/html-pdf/ETC2/DHT22.html	28
2.22 IDE do arduino. Fonte: https://learn.sparkfun.com/tutorials/what-is-an-arduino/all	29
2.23 Arduino Uno R3. Fonte: https://nettigo.eu/products/uno-r3-clone-atmega328p-mega16u2	30
2.24 Esquema eléctrico do arduino uno R3. Fonte: https://docs.arduino.cc/hardware/uno-rev3/tech-specs	31
2.25 ESP32. Fonte: https://victorvision.com.br/blog/placa-esp32/	32
2.26 Pinagem do esp32. Fonte: https://curtocircuito.com.br/blog//conhecendo-esp32	32
2.27 Módulo Relé. Fonte: https://www.mactronica.com.co/modulo-rele-1-canal	34
3.1 Esquema eléctrico do medidor. Fonte: Autor	39
3.2 Esquema eléctrico do sistema de controle. Fonte: Autor	41
3.3 Diagrama de blocos do sistema. Fonte: Autor	44
3.4 Servidor web ESP32 como uma estação WIFI. Fonte: https://linuxhint.com/esp32-wifi-arduino-ide/	45
4.1 Protótipo do sistema . Fonte: Autor	55
4.2 Valores de Corrente (A) e Potência (kW) no Display LCD. Fonte: Autor	56
4.3 Estado do cômodo com a temperatura abaixo do valor de acionamento: (a) Valor da temperatura, (b) Ventilador desligado. Fonte:Autor	57
4.4 Sensor DHT22 sendo aquecido com base em um ferro de passar roupa. Fonte: Autor	58
4.5 A temperatura ultrapassou o limite e por isso o ventilador é acionado: (a) Ventilador acionado, (b) Valor de temperatura acima da temperatura limite. Fonte:Autor	59

4.6	Lâmpada apagada pois não foi detectado movimento: (a) Lâmpada apagada, (b) Visualização de movimento ausente no Display LCD. Fonte:Autor	60
4.7	Lâmpada acesa pois foi detectado movimento: (a) Lâmpada acesa, (b) Visualização de movimento detectado no Display LCD. Fonte:Autor	61
4.8	Valores de Tensão (V), Corrente (A), Potência (kW), Consumos acumulado e limite (kWh) na página web. Fonte: Autor	62
4.9	Definindo o valor de 0.006kWh como Energia limite. Fonte: Autor	63
4.10	Notificação 1: Consumo acumulado atingiu 75% da Energia disponível. Fonte: Autor	63
4.11	Notificação 2: Consumo acumulado atingiu o valor da Energia disponível. Fonte: Autor	64

Lista de Tabelas

- 2.1 Consumo final de electricidade por sector, 2021. Fonte: [(Feldmann et al., 2023)] 17
- 2.2 Tarifas de electricidade para diferentes usos em baixa tensão. Fonte: [(Feldmann et al., 2023)] 17
- 2.3 Descrição dos Pinos do Sensor de Corrente ACS712. Fonte: <https://www.usinainfo.com.br/bmedindo-corrente-com-o-sensor-ac712-e-o-arduino/> 22

- 3.1 Componentes do subsistema de medição. Fonte: Autor 40
- 3.2 Componentes do subsistema de controle. Fonte: Autor 42

Lista de Acrónimos

A *Ampère*. 6

ACS712 *Sensor de corrente da Allegro 712 - Allegro Current Sensor 712*. 21

ADC *Conversor Analógico-Digital - Analog-to-Digital Converter*. 31

ADC-SAR *Conversor Analógico-Digital com Registro de Aproximação Sucessiva - Analog-to-Digital Converter – Successive Approximation Register*. 31

BLE *Bluetooth de Baixo Consumo de Energia - Bluetooth Low Energy*. 31

CA *Corrente Alternada*. xxiv

CC *Corrente Contínua*. 7

DHT22 *Umidade e temperatura digitais 22 - Digital humidity and temperature 22*. 27

EDM *Electricidade de Moçambique*. 6

EP *Empresa Pública*. 6

ESP-IDF *Espressif IoT Development Framework*. 32

ESP32 *Espressif Systems Platform 32*. 31

GND *Terra - Ground*. 23

GPIO *Entrada/Saída de Propósito Geral - General Purpose Input/Output*. 31

h *Hora*. 12

Hz *Hertz*. 8

I2C *Circuito Inter-Integrado - Inter-Integrated Circuit*. 31

ICSP *Programação Serial em Circuito - In-Circuit Serial Programming*. 30

IDE *Ambiente de Desenvolvimento Integrado - Integrated Development Environment.* 29

IoT *Internet das coisas - Internet of Things.* 31

J *Joules.* 12

JSON *Notação de Objeto JavaScript - JavaScript Object Notation.* 46

KB *Kilo-Byte.* 30

kW *Quilowatt.* 12

kWh *Quilowatt-hora.* 12

LCD *Visor de Cristal Líquido - Liquid Crystal Display.* 13

LED *Diodo Emissor de Luz - Light-Emitting Diode.* 13

mA *mili-Ampère Volts Alternating Current.* 24

MB *Mega-Byte.* 31

MHz *Mega-Hertz.* 30

MWh *Megawatt-hora.* 16

PBT *Tereftalato de Polibutileno - Polybutylene Terephthalate.* 55

PIR *Infravermelho Passivo - Passive infrared.* 25

PWM *Modulação por Largura de Pulso - Pulse Width Modulation.* 30

RMS *Raiz Média Quadrática - Root Mean Square.* 47

RTC *Relógio em Tempo Real - Real-Time Clock.* 33

s *Segundos.* 8

SCL *Linha de Clock Serial - Serial Clock Line.* 33

SDA *Linha de Dados Serial - Serial Data Line.* 33

SoC *Sistema em um Chip - System on a Chip.* 31

SPI *Interface Serial Periférica - Serial Peripheral Interface.* 31

SRAM *Memória Estática de Acesso Aleatório - Static Random Access Memory.* 30

SSID *Identificador do Conjunto de Serviço - Service Set Identifier.* 46

TI *Tecnologia da Informação.* 1

UART *Transmissor/Receptor Assíncrono Universal - Universal Asynchronous Receiver/Transmitter.*
31

USB *Barramento serial universal - Universal Serial Bus.* 30

V *Volt.* 6

VA *Volt-Ampère.* 11

VAC *Volts de Corrente Alternada - Volts Alternating Current.* 24

VA_r *Volt-Ampère Reativo.* 11

VCC *Tensão de Coletor Comum - Voltage Common Collector.* 22

VDC *Volts de Corrente Contínua - Volts Direct Current.* 24

W *Watts.* 6

WEB *Rede Mundial de Computadores - World Wide Web.* 46

WiFi *Wireless Fidelity.* 31

Capítulo 1

Contextualização do tema

1.1 Introdução

A energia eléctrica é uma das formas de energia mais utilizadas no mundo, ela desempenha um papel essencial no desenvolvimento das sociedades modernas. Ela é um recurso vital para praticamente todas as actividades das mais variadas naturezas: a maior parte da população utiliza energia em suas residências diariamente para tarefas simples do dia a dia; empresas e indústrias à utilizam na produção de bens e serviços; por sua vez, o setor público necessita de energia para viabilizar o funcionamento de serviços básicos, como iluminação pública e atendimento médico, por exemplo. Em Moçambique, além da energia eléctrica ocupar um papel essencial em todos os sectores, ela impacta na balança comercial pois o país gera grandes quantidades de energia tanto para o consumo interno quanto para exportação, enviando eletricidade para diversos países da África Austral.

A implementação de tecnologias para medição e controle do consumo de energia eléctrica têm um impacto significativo na economia de energia, na redução de custos e na sustentabilidade ambiental. Elas permitem identificar padrões de consumo, detectar alterações na rede eléctrica, evitar desperdícios e melhorar a eficiência operacional. Além disso, contribuem para a diminuição das emissões de gases poluentes e criação de novas oportunidades de emprego nas áreas de Engenharia, *Tecnologia da Informação* (TI) e automação.

O desperdício de energia eléctrica é um problema cada vez mais recorrente que afeta não apenas os consumidores residenciais, empresas e instituições, mas também os sistemas de geração e distribuição de energia em larga escala. Trata-se de uma questão complexa que vai além do simples aumento de custos, pois impacta na economia, no

meio ambiente e na estabilidade dos sistemas energéticos. Além disso, o consumidor não tem informações sobre os parâmetros da rede eléctrica o que representa um grande obstáculo para o uso consciente e eficiente da energia eléctrica, além de aumentar a vulnerabilidade do consumidor à falhas na rede.

O presente trabalho tem como objetivo propor um sistema para medir e controlar o consumo de energia eléctrica para o consumidor. Para isso, foram analisados os métodos de medição e controle do consumo de energia, identificou-se os requisitos e posteriormente projectou-se o sistema.

1.2 Formulação do problema

O consumo de energia eléctrica é um dos aspectos fundamentais da vida moderna, impactando directamente no bem-estar das pessoas, no desenvolvimento económico e no meio ambiente, ou seja, ele funciona como um termómetro da atividade produtiva e do bem-estar da população. Em Moçambique, o consumo de energia eléctrica tem evoluído significativamente nos últimos anos, refletindo os esforços contínuos para expandir o acesso à electricidade. No entanto, o desperdício de energia eléctrica é um desafio enfrentado diariamente por milhares de consumidores, especialmente no uso de sistemas de iluminação e climatização. Na iluminação, a principal causa do desperdício é deixar luzes acesas em ambientes que não estão sendo utilizados, no sistema de climatização, ventiladores e ar-condicionados são usados sem o devido controle de temperatura e muitas das vezes em ambientes vazios. Além do desperdício de energia eléctrica por parte do consumidor, tem se verificado por vezes a ocorrência de falhas na rede eléctrica tais como sobretensões e picos de corrente, essas alterações frequentemente passam despercebidas para os consumidores pois não tem acesso à informações que permitem identificá-las. O desperdício de energia eléctrica agravado pelo facto da tarifa média de electricidade no país ser elevada e continuar a aumentar resulta em maiores gastos com electricidade para as residências o que reduz o poder de compra para outros bens e serviços e redução das actividades ou encerramento de alguns sectores em algumas empresas e instituições. O desperdício de energia também resulta no desgaste prematuro dos aparelhos de iluminação e ventilação pelo seu uso constante. Somado a isso, alterações na rede eléctrica se não identificadas com antecedência podem causar danos aos equipamentos eléctricos e electrónicos e comprometer a segurança e o bem estar do

consumidor. Desta forma surge a seguinte questão de pesquisa: Que tecnologias podem ser usadas de forma a reduzir o desperdício de energia eléctrica e detectar alterações na rede eléctrica?

1.3 Justificativa

Desenvolver um sistema de leitura e controle do consumo de energia eléctrica é de grande relevância em um mundo cada vez mais preocupado com a sustentabilidade, a economia e a eficiência. Os impactos causados por esse sistema podem ser observados sob diferentes perspectivas, incluindo a do usuário, da sociedade em geral e do próprio desenvolvedor da tecnologia.

Para o usuário haverá uma redução significativa nos custos com a conta de energia permitindo-lhe economizar dinheiro. O acesso a dados além do consumo em tempo real, permitirá ao usuário conhecer o seu padrão de consumo e identificar alguns problemas com antecedência e desta maneira resolver e/ou minimizar os seus prejuízos.

Do ponto de vista da sociedade, haverá redução da demanda energética geral, diminuindo a necessidade de expansão de usinas e redes eléctricas, reduzindo os custos públicos e a pressão sobre os recursos naturais. A sociedade terá um consumo racional de energia o que ajudará na diminuição das emissões de gases poluentes e na diminuição da dependência de fontes poluentes como termoeléctricas.

Para o pesquisador, este projecto o permitirá trabalhar com tecnologias modernas o que enriquecerá o seu conhecimento técnico e aumentará o seu valor no mercado de trabalho. Além disso, há um mercado em crescimento para esse tipo de solução o que pode gerar reconhecimento profissional e oportunidades de empreendedorismo.

1.4 Objectivos

1.4.1 Objectivo Geral

Propor um sistema de medição e controle do consumo de energia eléctrica.

1.4.2 Objectivos Específicos

- a. Analisar as formas de medição e de controle do consumo de energia eléctrica;

- b. Identificar os requisitos para implementar um sistema de medição e de controle do consumo de energia eléctrica;
- c. Projectar o subsistema de medição do consumo de energia;
- d. Projectar o subsistema de controle do consumo de energia.

1.5 Metodologia de investigação

A metodologia de investigação é um sistema de princípios e regras que orienta o processo de uma investigação. Nesta secção é apresentada a classificação da pesquisa e as ferramentas usadas para a elaboração do projecto.

1.5.1 Classificação da Pesquisa

1.5.1.1 Quanto à natureza

A presente investigação caracteriza-se como uma pesquisa aplicada, pois visa propor uma solução prática para um problema concreto.

1.5.1.2 Quanto à abordagem

Quanto ao tipo de abordagem, trata-se de uma pesquisa de natureza qualitativa, com apoio de dados quantitativos sempre que necessário. A abordagem qualitativa será predominante na análise de documentos, revisão bibliográfica e elaboração da proposta do sistema, enquanto os dados quantitativos poderão ser utilizados na simulação e avaliação de desempenho.

1.5.1.3 Quanto aos objectivos

É uma pesquisa exploratória, uma vez que se busca aprofundar o conhecimento sobre métodos, tecnologias e práticas existentes nesta área, sem a pretensão de esgotar o tema.

1.5.1.4 Quanto à técnica de pesquisa

O presente trabalho é do tipo pesquisa bibliográfica porque foi baseado em livros, artigos, revistas publicadas, websites e datasheet.

1.5.2 Ferramentas Utilizadas

- a. **Arduino IDE** usado para escrever e fazer upload de programas nos microcontroladores;
- b. **EasyEDA** usado para desenhar os esquemas eléctricos de cada subsistema;
- c. **LucidChart** usado para desenhar o diagrama do sistema;
- d. **Latex** usado para documentação e edição.

1.6 Estrutura do trabalho

O presente trabalho está dividido em cinco capítulos de desenvolvimento do texto apresentados a seguir:

- **CAPÍTULO 1** – Nesse capítulo é apresentada a introdução e a formulação do problema, inclui também os objectivos, justificativa e metodologias que foram levantadas para a investigação.
- **CAPÍTULO 2** – Este é o capítulo da fundamentação teórica, onde os conceitos relevantes e que fizeram parte da pesquisa são detalhados.
- **CAPÍTULO 3** – Neste capítulo, é apresentada e descrita a proposta de solução, além de ser explorado o desenvolvimento do sistema.
- **CAPÍTULO 4** – Os ensaios e resultados obtidos são apresentados e discutidos neste capítulo.
- **CAPÍTULO 5** – Apresentação sintética dos resultados da pesquisa e das soluções.

Capítulo 2

Revisão Teórica

2.1 Energia Eléctrica

Segundo Guedes (2011, p.08), “ Energia eléctrica é o produto da potência eléctrica pelo intervalo de tempo de utilização de um equipamento ou de funcionamento de uma instalação (residencial, comercial ou industrial)”. Em Moçambique, a energia eléctrica é fornecida pela *Electricidade de Moçambique* (EDM), *Empresa Pública* (EP) responsável pelo sector eléctrico no país. A EDM foi fundada em 1977 e abrange a geração, transmissão, distribuição e comercialização de energia eléctrica.

2.1.1 Análise da energia eléctrica em CA

Um circuito eléctrico é um conjunto de componentes eléctricos e electrónicos conectados entre si e que podem ser percorridos por uma corrente eléctrica. Pode-se definir corrente eléctrica como o fluxo ordenado de cargas eléctricas que ocorre devido à diferença de potencial entre dois pontos do circuito que é chamada de tensão eléctrica. A unidade de medida da corrente eléctrica é o *Ampère* (A) e a unidade da tensão eléctrica é o *Volt* (V). A medida da energia eléctrica consumida ou fornecida pelo circuito é chamada de potência eléctrica. Essa medida é essencial para compreender o funcionamento dos dispositivos eléctricos. Em outras palavras, é a quantidade de trabalho que a electricidade realiza por unidade de tempo. Quanto maior a potência, mais energia eléctrica está sendo consumida ou fornecida por um dispositivo eléctrico em um determinado período. A potência eléctrica é medida em *Watts* (W) e é dada a partir da seguinte fórmula:

$$P = V \times I \quad (2.1)$$

2.1.1.1 Características da CA

Para os autores Segundo e Rodrigues (2015), a CA é um tipo de corrente eléctrica em que tanto a intensidade quanto a direcção mudam de forma cíclica ao longo do tempo. Isso é diferente da *Corrente Contínua* (CC) que tem uma direcção constante e não sofre variações ao longo do tempo. Em resumo, na CC, os electrões fluem em apenas um sentido, enquanto na CA, o fluxo de electricidade alterna entre dois sentidos. A CA foi adoptada para o transporte e distribuição da energia eléctrica devido à facilidade de elevação ou diminuição do valor de sua tensão alternada por intermédio de transformadores. Abaixo são apresentadas as suas principais características:

Valor Instantâneo - $u(t)$

O valor instantâneo de uma grandeza alternada sinusoidal - u - pode representar-se matematicamente em função do tempo - t :

$$u(t) = U_m \sin(\omega t) \quad (2.2)$$

onde ω indica a velocidade angular, ou seja, a variação do ângulo de um sinal sinusoidal em função do tempo. Essa grandeza é expressa em radianos por segundo (rad/s). A velocidade angular está relacionada com a frequência e o período através da seguinte equação:

$$\omega = 2\pi f = \frac{2\pi}{T} \quad (2.3)$$

Se considerarmos um vector U , de comprimento U_m , girando à velocidade ω , o valor instantâneo u corresponde à projecção vertical desse vector (Alves, 1999):

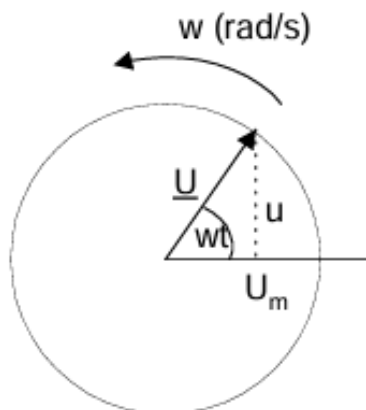


Figura 2.1: Projecção de um vector. Fonte: [(Alves, 1999)]

Período - T e Frequência - f

Como a CA se repete de forma cíclica, uma de suas principais características é o intervalo de tempo que o sinal leva para completar um ciclo, chamado de período T, medido em *Segundos* (s).

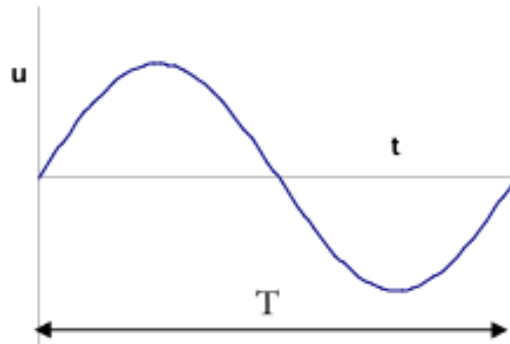


Figura 2.2: Representação gráfica de período. Fonte: [(Alves, 1999)]

Outra característica frequentemente usada na CA é a frequência, que está directamente ligada ao período. A frequência é a medida do número de ciclos completos que ocorrem em um segundo, sendo medida em *Hertz* (Hz). A relação entre a frequência e o período é a seguinte:

$$f = \frac{1}{T} \quad (2.4)$$

É importante destacar que tanto o período quanto a frequência são propriedades comuns a qualquer sinal periódico. Isso significa que esses conceitos não se aplicam apenas à CA sinusoidal, mas também à outros tipos de sinais, como os de forma quadrada, triangular, digital, entre outros (Alves, 1999).

Amplitude Máxima - Um

A amplitude máxima, também conhecida como valor de pico ou valor máximo, corresponde ao maior valor instantâneo que um sinal pode atingir, tanto no sentido positivo como no sentido negativo. No caso da tensão e da corrente, essa amplitude é representada pelos U_m e I_m , respectivamente.

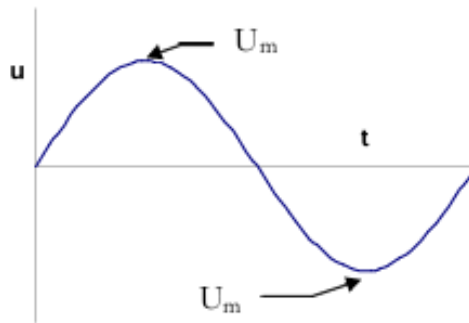


Figura 2.3: Representação gráfica de amplitude máxima. Fonte: [(Alves, 1999)]

O valor eficaz de uma grandeza alternada corresponde ao valor de uma grandeza contínua que, aplicada à mesma resistência durante o mesmo intervalo de tempo, faria que ela dissipasse a mesma quantidade de calor que a alternada. Em outras palavras, o valor eficaz é uma maneira de comparar a produção de trabalho entre sistemas de CA e de CC.

Para grandezas alternadas com forma de onda sinusoidal, o valor eficaz é 2 vezes menor que o valor máximo, independentemente da frequência :

$$I = \frac{I_m}{\sqrt{2}}; U = \frac{U_m}{\sqrt{2}} \quad (2.5)$$

2.1.1.2 POTÊNCIAS INSTANTÂNEA, ACTIVA, REACTIVA E APARENTE

Potência Instantânea

Considere-se um circuito ao qual se aplicou uma tensão:

$$u(t) = U_m \sin(\omega t) \quad (2.6)$$

e que é percorrido pela corrente:

$$i(t) = I_m \sin(\omega t + \Phi) \quad (2.7)$$

A potência instantânea dissipada em um determinado momento é obtida pelo produto entre a tensão u e a corrente i . Vamos agora apresentar o gráfico da potência instantânea p para diferentes tipos de circuitos. Para cada instante de tempo, multiplicam-se os valores de u e i , levando em consideração o sinal algébrico de acordo com o sentido das grandezas. Supondo que os valores máximos de tensão e corrente sejam $U_m = 1,5V$ e

$I_m = 1A$, podemos representar graficamente as grandezas de corrente, tensão e potência em função do tempo (Alves, 1999).

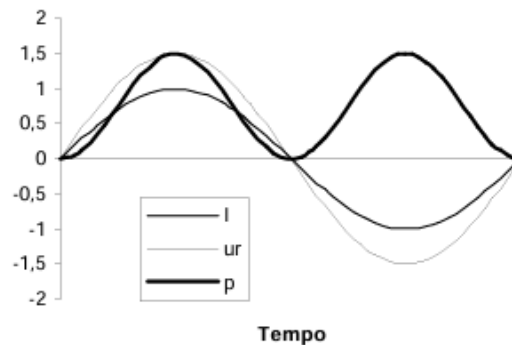


Figura 2.4: Gráfico de potência instantânea para diferentes circuitos. Fonte: [(Alves, 1999)]

O fato de a potência p ser sempre positiva indica que o circuito está absorvendo energia, a qual está sendo dissipada na resistência.

Potência Activa

Em certos momentos, a potência é nula, ou seja, a resistência não recebe energia; em outros, ela atinge seu valor máximo. No entanto, na prática, o que realmente importa é o valor médio dessa potência (P), que corresponde, no gráfico da Figura acima, ao valor médio da curva senoidal de p (Alves,1999).

$$I = \frac{U_m I_m}{2} = \frac{\sqrt{2}U \sqrt{2}I}{2} = UI \quad (2.8)$$

Esta potência média é a potência activa medida pelos Wattímetros (aparelhos de medida de potência). A sua expressão geral é:

$$P = RI^2 = UI \cos \Phi \quad (2.9)$$

em que Φ é o ângulo entre a tensão e a corrente (no caso da resistência, $\Phi = 0^\circ$ e $\cos 90^\circ = 1$)

Potência Reactiva

Podemos também traçar o gráfico da potência instantânea para uma indutância pura,

considerando os mesmos valores máximos para a tensão e corrente :

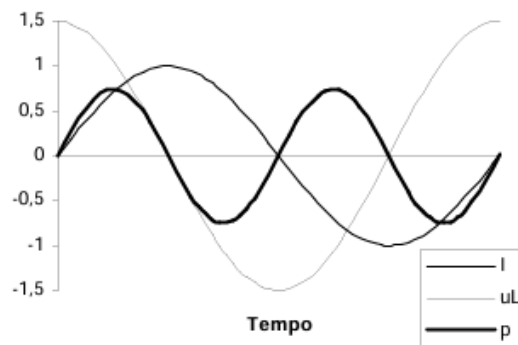


Figura 2.5: Gráfico de potência instantânea para indutância pura. Fonte: [(Alves, 1999)]

importante observar que a potência instantânea p varia entre valores positivos e negativos, com uma frequência que é o dobro da da tensão e da corrente na indutância. Quando essa potência é positiva, o receptor está a consumir energia da fonte; quando é negativa, está a devolver energia à fonte. No caso de uma indutância, há uma troca constante de energia — ora recebe, ora devolve — de forma que, no total, a energia média trocada é nula, ou seja, a energia recebida é igual à devolvida, sem perdas por dissipação. Se utilizarmos um wattímetro para medir a potência activa, ele indicará zero, ou seja, $P=0W$. Apesar de essa energia não ser efetivamente consumida, ela circula pelo circuito sob a forma de corrente eléctrica. A potência associada a essa energia oscilante chama-se Potência Reactiva, representada por Q (Alves, 1999).

No caso geral, para determinarmos a potência aparente de um elemento ou circuito, utilizamos a seguinte expressão:

$$Q = UI \sin \Phi \quad (2.10)$$

Onde U e I representam, respetivamente, a tensão e a corrente no elemento ou circuito, e Φ é o ângulo de defasagem entre elas. Para uma indutância ideal, esse ângulo é de 90° (sendo $\sin 90^\circ=1$). A potência reactiva pode ser medida com o uso de varímetros, e a sua unidade é o *Volt-Ampère Reactivo* (VAr) (Alves, 1999).

Potência Aparente

Potência aparente refere-se à potência total que uma determinada fonte é capaz de fornecer a um sistema. Representa-se pela letra S , é medida em *Volt-Ampère* (VA) e pode

ser calculada através da seguinte fórmula:

$$S = UI \quad (2.11)$$

Em termos vectoriais, podemos representar o chamado triângulo de potências (caso indutivo):

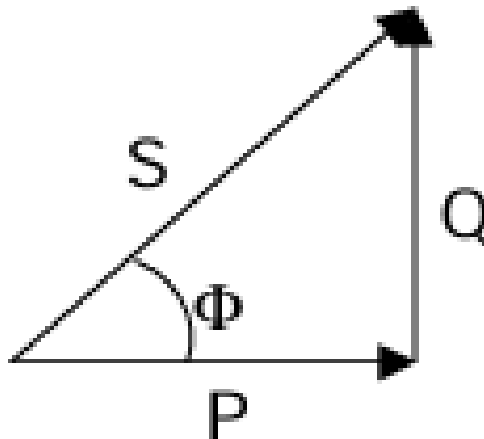


Figura 2.6: Triângulo de potências. Fonte: [(Alves, 1999)]

Podemos então relacionar o módulo das três potências da seguinte maneira:

$$S = \sqrt{P^2 + Q^2} \quad (2.12)$$

2.1.2 Medidores de Energia Eléctrica

A medição de energia eléctrica consiste em determinar a quantidade de electricidade utilizada por um aparelho, instalação, edifício ou sistema. A unidade mais utilizada para essa medição é o *Quilowatt-hora* (kWh), que representa a energia consumida por uma carga de 1 *Quilowatt* (kW) ao longo de uma *Hora* (h).

O medidor de energia, popularmente chamado de "relógio de luz", realiza essa medição de forma contínua, registrando os valores instantâneos de tensão em V e corrente eléctrica em A para calcular a energia consumida, expressa em unidades como *Joules* (J) ou kWh. Atualmente, os medidores de energia eléctrica se classificam em dois tipos principais: electromecânicos e electrónicos.

2.1.2.1 Medidores Electromecánicos

Os medidores electromecánicos, conhecidos como os modelos tradicionais de relógios com disco, funcionam por meio da passagem de corrente eléctrica por suas bobinas, o que gera um campo magnético. Esse campo faz com que um disco metálico dentro do aparelho gire apenas quando há consumo de energia. Esse movimento aciona engrenagens e ponteiros que registram o consumo. O giro do disco é proporcional à intensidade do campo magnético: quanto maior o consumo de energia, mais intenso o campo e mais rápido o disco gira. Embora sejam dispositivos confiáveis e com baixo custo de fabricação, esses medidores geralmente exigem que a leitura seja feita manualmente.



Figura 2.7: Medidor Electromecánico Fonte: https://pt.wikipedia.org/w/index.php?title=Medidor_de_energia_el%C3%A9trica&action=info

2.1.2.2 Medidores Electrónicos

Os medidores electrónicos realizam a medição de energia de forma digital, utilizando tecnologias mais avançadas em comparação aos modelos tradicionais. Eles apresentam as informações de consumo em visores *Visor de Cristal Líquido - Liquid Crystal Display* (LCD) ou *Diodo Emissor de Luz - Light-Emitting Diode* (LED) e, em muitos casos, pos-

suem a capacidade de enviar os dados de leitura automaticamente para centrais remotas. Além de registrar o consumo de energia eléctrica, esses medidores também podem monitorar outros aspectos do fornecimento e da carga eléctrica, como a demanda instantânea e máxima, valores de tensão, fator de potência, potência reativa, entre outros. Por conta dessas funcionalidades adicionais, são amplamente utilizados em instalações comerciais e industriais.



Figura 2.8: Medidor Electrónico. Fonte: https://pt.wikipedia.org/w/index.php?title=Medidor_de_energia_el%C3%A9trica&action=info

2.1.3 Classificação de cargas eléctricas

Conforme Vasconcelos (2017), a carga eléctrica refere-se à potência absorvida por um equipamento ou por um conjunto de equipamentos a partir de uma fonte de suprimento. Em um sentido mais amplo, o próprio equipamento pode ser denominado carga. Diferentes componentes do sistema eléctrico podem ser considerados cargas, como consumidores individuais, transformadores, alimentadores primários e até subestações. As cargas podem ser classificadas segundo diversos critérios, sendo os mais comuns:

- a. Localização geográfica: conforme a posição do consumidor, as cargas podem ser urbanas, suburbanas ou rurais.

- b. Sensibilidade ao suprimento: algumas cargas são mais dependentes da continuidade no fornecimento de energia, sendo classificadas como:
 - i. Sensíveis: qualquer interrupção, mesmo que breve, pode causar grandes prejuízos (ex.: hospitais, indústrias de processos contínuos etc.).
 - ii. Semi-sensíveis: interrupções de até 10 minutos não geram impactos tão severos quanto nas cargas sensíveis (ex.: comércios, prédios residenciais etc.).
 - iii. Normais: interrupções ocasionais não afetam significativamente o funcionamento (ex.: residências, pequenos estabelecimentos comerciais etc.).
- c. Pagamento ou rentabilidade: classificação dos consumidores em pequenos, médios e grandes, com base no faturamento da concessionária.
- d. Grau de dependência do suprimento: cargas que sofrem impacto significativo com variações de tensão, como fornos a arco e processos automatizados em fábricas.
- e. Tipo de atividade: de acordo com o uso da energia, podendo ser industrial, comercial, residencial ou de serviço público. Essa classificação também influencia a definição das tarifas aplicadas.
- f. Ciclo de trabalho: classificadas como transitórias cíclicas, transitórias acíclicas ou contínuas, dependendo do seu padrão de funcionamento ao longo do tempo.

A carga pode ser representada graficamente ao longo do tempo por meio da curva de carga, que pode ser diária, semanal, mensal ou anual, dependendo do intervalo analisado. Cada consumidor apresenta uma curva de carga característica que permite classificá-lo adequadamente. Por exemplo, um consumidor residencial típico (Figura 2.9) tem maior consumo entre 17h e 22h, com um pico entre 19h e 21h, devido ao uso intensivo da iluminação. Já em cargas comerciais (Figura 2.10), a demanda começa a aumentar entre 8h e 9h, acompanhando o início das atividades, e sofre uma queda acentuada após 18h, quando os estabelecimentos encerram o expediente (Vasconcelos, 2017).

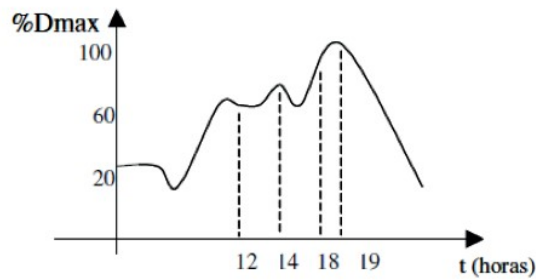


Figura 2.9: Curva típica de carga residencial. Fonte:[(Vasconcelos, 2017)]

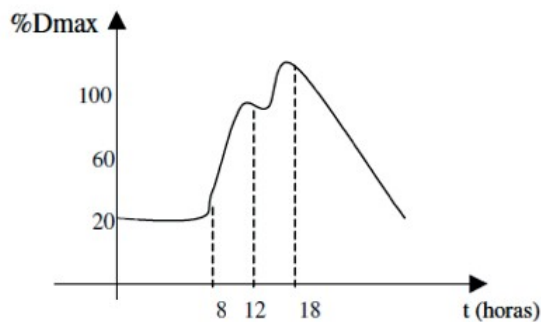


Figura 2.10: Curva típica de carga comercial. Fonte:[(Vasconcelos, 2017)]

2.1.4 Consumo e tarifas de electricidade

Consumo de energia eléctrica é a quantidade de potência eléctrica kW consumida em um intervalo de tempo, expresso em kWh ou em pacotes de 1000 unidades *Megawatt-hora* (MWh). Para um equipamento eléctrico, o valor é determinado pelo produto entre sua potência e o tempo de utilização. Já em uma instalação residencial, comercial ou industrial, esse valor é obtido pela soma dos produtos da demanda medida pelo respectivo período de integração (Souza Guedes, 2011).

$$Consumo = \frac{Potencia(W)}{1000} \times tempo(h) = Total(kWh) \quad (2.13)$$

Demanda é média das potências eléctricas ativas ou reativas requisitadas ao sistema eléctrico pela parte da carga instalada em operação na unidade consumidora, ao longo de um determinado intervalo de tempo. Por outro lado, demanda contratada é a demanda de potência activa que a concessionária deve fornecer de forma contínua e obrigatória no ponto de entrega, conforme os termos do contrato de fornecimento. Esse valor, estabelecido em contrato por um período específico, deve ser pago integralmente pelo consumi-

dor, independentemente do uso durante o período de faturamento. A unidade de medida utilizada é o kW (Souza Guedes, 2011).

A tabela seguinte apresenta os dados do consumo de electricidade em Moçambique no ano de 2021. O sector mais consumidor é o doméstico, seguido da indústria e do comércio.

Tabela 2.1: Consumo final de electricidade por sector, 2021. Fonte: [(Feldmann et al., 2023)]

Sector	Porcentagem
Doméstico	44.9 %
Industrial	37.5 %
Comercial	15.0 %
Agricultura	0.9 %
Iluminação pública	1.4 %
Consumo da EDM	0.2 %

A EDM oferece diferentes opções tarifárias (tarifa é o preço da unidade de energia eléctrica). A tabela 2.3 seguinte mostra os preços actuais (em MZN/kWh) para a utilização de baixa tensão para as famílias e pequenos agricultores. Estas tarifas são uniformes em todo o país .

Tabela 2.2: Tarifas de electricidade para diferentes usos em baixa tensão. Fonte: [(Feldmann et al., 2023)]

Consumo registado(kWh)	Tarifa Social(Mt/kWh)	Tarifas domésticas(Mt/kWh)	Tarifa agrícola (Mt/kWh)	Tarifa Geral(Mt/kWh)	Taxa fixa (Mt/kWh)
De 0 a 100	0.97				
De 0 a 200		6.00	3.69	9.32	233.37
De 201 a 500		8.49	5.26	13.31	233.37
Acima de 500		8.91	5.75	14.56	233.37
Pré-pagamento	0.97	7.64	5.11	13.34	

Factura de energia eléctrica é um documento fiscal que detalha o valor total a ser pago pelo fornecimento de energia eléctrica em um período específico, apresentando a discriminação das respectivas parcelas. Um dos métodos de faturação disponibilizados aos clientes da EDM é por meio de contadores eléctricos convencionais. Esses dispositivos

são cedidos aos consumidores e lidos mensalmente por funcionários treinados da EDM, que calculam o consumo de electricidade do período. O pagamento das faturas pode ser feito por meio de depósitos bancários, preenchimento de faturas nos bancos, transferências via multibanco ou débito automático em contas de folha de pagamento. Como alternativa, há o sistema pré-pago CREDELEC, que permite aos consumidores controlar seus gastos com electricidade em um determinado período, sem a necessidade de pagar uma taxa mensal fixa. Esse modelo evita cortes inesperados no fornecimento por inadimplência. As recargas do CREDELEC podem ser adquiridas por meio de cartões de raspadinha, disponíveis em lojas de conveniência e supermercados, ou diretamente nos ATMs da EDM. Em 2020, aproximadamente 78% dos consumidores de energia eléctrica em Moçambique (mais de 671.322 clientes) utilizavam o sistema CREDELEC para faturação (Feldmann et al., 2023).

2.2 Sistemas de controle

Segundo Nise (2013, p.02), "Um sistema de controle consiste em subsistemas e processos (ou plantas) construídos com o objectivo de se obter uma saída desejada com um desempenho desejado, dada uma entrada especificada".

O uso de sistemas de controle automático está amplamente presente no cotidiano das sociedades avançadas. Esses sistemas desempenham um papel crucial no impulso ao progresso e ao desenvolvimento. O controle automático é uma parte essencial e indispensável de sistemas como os de veículos espaciais (como o acionamento de foguetes e a decolagem de ônibus espaciais para orbitar a Terra), sistemas robóticos, tecnologias de manufactura modernas e diversas operações industriais que envolvem o controle de parâmetros como temperatura, pressão, humidade, viscosidade, vazão, entre outros. A figura a seguir ilustra um sistema de controle em sua versão mais básica, onde a entrada corresponde a uma saída desejada.

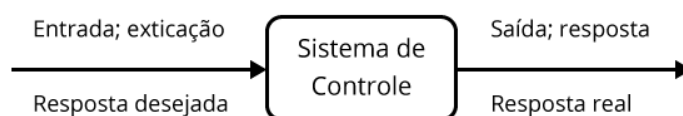


Figura 2.11: Descrição simplificada de um sistema de controle. Fonte: [(Nise, 2013)]

2.2.1 Configurações dos sistemas de controle

Existem duas principais configurações dos sistemas de controle: malha aberta e malha fechada.

2.2.1.1 Sistemas de controle de malha aberta

Sistemas de controle em malha aberta são aqueles nos quais a saída não influencia directamente no funcionamento do sistema. Isso significa que a resposta do sistema não é monitorada nem usada para ajustar a entrada. Um exemplo comum é o forno de micro-ondas: ao seleccionar a função “descongelar”, o aparelho executa o processo com base em um tempo previamente definido, sem verificar se o alimento foi realmente descongelado como esperado. É necessário abrir o forno e verificar manualmente. As principais vantagens desse tipo de sistema são a estrutura simples e o custo limitado. No entanto, a ausência de realimentação pode levar a resultados imprecisos.

Um sistema em malha aberta é mostrado na figura seguinte. Ele se inicia com um componente chamado de transdutor de entrada, que transforma o sinal de entrada para um formato compatível com o controlador. O controlador, por sua vez, atua sobre o processo ou planta. A entrada do sistema, também conhecida como referência, é o valor desejado, enquanto a saída é chamada de variável controlada. Além disso, podem existir sinais externos, chamados de perturbações, que interferem tanto na saída do controlador quanto ao processo. Esses sinais são combinados por meio de pontos de soma, que realizam a adição algébrica dos valores recebidos (Nise, 2013).

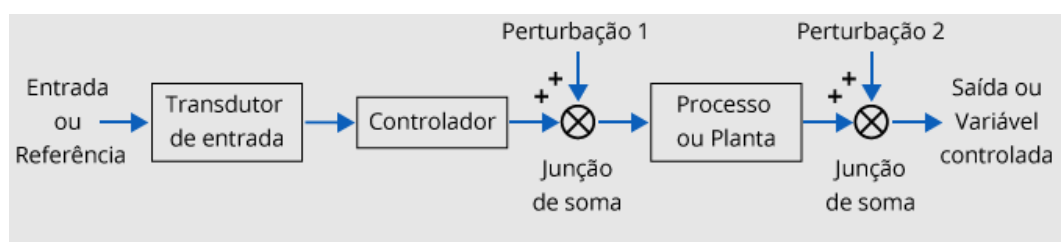


Figura 2.12: Diagrama de blocos de um sistema de controle em malha aberta.

Fonte: [(Nise, 2013)]

2.2.1.2 Sistemas de controle de malha fechada

O termo “malha fechada” está sempre relacionado ao uso da realimentação para minimizar erros. Entre as principais vantagens desse tipo de sistema estão a correção automática

de desvios, estabilidade na saída e maior resistência a interferências externas. Por outro lado, esse sistema tende a ser mais complexo e mais caro.

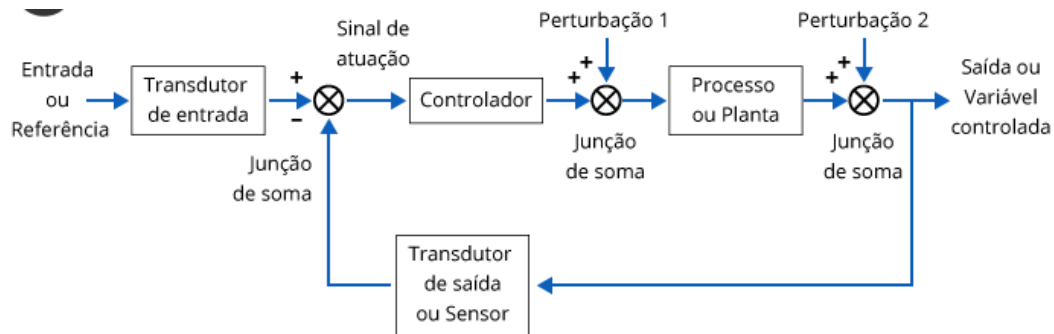


Figura 2.13: Diagrama de blocos de um sistema de controle em malha fechada. Fonte:[(Nise, 2013)]

A representação de um sistema em malha fechada é mostrada na figura acima. O transdutor de entrada transforma o sinal de entrada em um formato que o controlador consegue interpretar. Já o transdutor de saída, ou sensor, capta a resposta do sistema e também a converte para um formato compatível pelo controlador. A primeira junção de soma adiciona algebricamente o sinal de entrada ao sinal de saída, que chega através da malha de realimentação, o caminho de retorno da saída para a junção de soma. O sinal de saída é subtraído do sinal de entrada. O resultado, geralmente, é chamado de sinal de atuação. Porém, quando os transdutores de entrada e saída possuem ganho unitário (ou seja, amplificam o sinal por um fator de 1), esse sinal de atuação corresponde diretamente à diferença entre a entrada e a saída. Nessa situação, ele é denominado erro. O sistema em malha fechada tem a capacidade de compensar perturbações externas ao monitorar continuamente a saída, realimentá-la e comparar com o sinal de entrada. Caso exista uma diferença entre os dois, o sistema gera um sinal de atuação que comanda a planta para realizar os ajustes necessários. Por outro lado, se não houver diferença, o sistema entende que a saída é correta e, portanto, não interfere (Nise, 2013).

2.2.2 Sistema de controle do consumo de energia

Um sistema de controle do consumo de energia é um conjunto de hardware e software, incluindo medidores e softwares de gestão, que monitoram o uso de energia em tempo real, identificam padrões, detetam anomalias e oferecem relatórios detalhados. O objetivo é aumentar a eficiência energética, reduzir custos, minimizar o impacto ambiental

e otimizar a gestão de recursos, sendo implementado tanto em residências como em empresas.

2.2.2.1 Componentes de um sistema de controle de energia

- a. **Medidores de energia:** dispositivos que medem o consumo e a qualidade da energia em diferentes circuitos, utilizando sensores para recolher dados sem necessidade de interromper a ligação eléctrica;
- b. **Software de gestão:** plataformas que se conectam aos medidores para coletar, processar e analisar os dados de consumo;
- c. **Plataformas de visualização:** Aplicações que apresentam os dados através de gráficos, dashboards e relatórios, permitindo ao utilizador monitorizar o uso de energia em tempo real.

2.3 Sensores

Um sensor é um dispositivo que reage a estímulos físicos ou químicos de forma específica, gerando um sinal que pode ser convertido em outra forma de grandeza física com o objectivo de realizar medições ou monitoramentos. Todos os sensores possuem componentes chamados transdutores, que têm a função de converter um tipo de energia em outro. Na maioria dos casos, esses transdutores transformam a grandeza recebida em um sinal eléctrico, que pode então ser analisado por sistemas eléctricos ou electrónicos. Os sensores podem ser classificados como analógicos, quando registram variáveis contínuas dentro de certos limites, ou digitais, quando detectam variáveis binárias, isto é, que só podem ter dois estados distintos.

2.3.1 Sensor de corrente invasivo

O sensor de corrente invasivo consiste em um circuito linear de efeito Hall *Sensor de corrente da Allegro 712 - Allegro Current Sensor 712 (ACS712)* de baixo desvio e alta precisão. Esse efeito, descoberto por Edwin H. Hall em 1879, ocorre quando uma corrente eléctrica percorre um condutor sujeito a um campo magnético perpendicular à direcção da corrente. Essa configuração provoca a deflexão dos portadores de carga (geralmente

electrões) para um dos lados do condutor, criando uma separação de cargas. Essa separação gera uma diferença de potencial transversal à corrente, conhecida como tensão Hall. A magnitude dessa tensão é proporcional à intensidade do campo magnético aplicado, à corrente eléctrica que flui pelo condutor e à densidade de portadores de carga no material. A tensão de operação do sensor é de 5 volts, ele pode detectar com precisão CA ou CC. A corrente máxima (CA ou CC) que pode ser detectada chega a 30A e o sinal de corrente presente pode ser lido pela porta analógica de entrada/saída do Arduino.

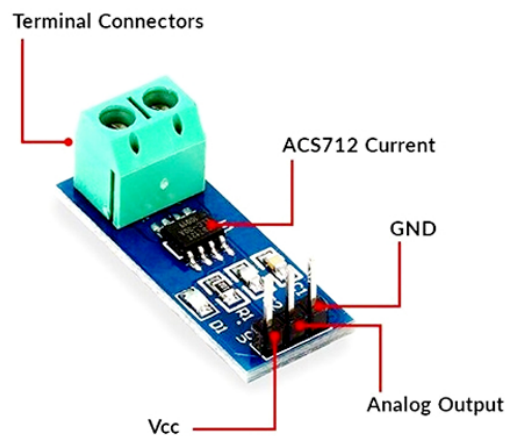


Figura 2.14: Sensor de corrente ACS712. Fonte: <https://www.usinainfo.com.br/blog/projeto-medindo-corrente-com-o-sensor-ac712-e-o-arduino/>

Tabela 2.3: Descrição dos Pinos do Sensor de Corrente ACS712. Fonte: <https://www.usinainfo.com.br/blog/projeto-medindo-corrente-com-o-sensor-ac712-e-o-arduino/>

Pino	Descrição
Vcc	A tensão de entrada é +5V para aplicações típicas.
Out	saída Tensão analógica proporcional à corrente.
Gnd	Conectado ao terra do circuito.
Terminal de conexão	O fio onde passa a ser medida deve ser conectado aqui.

A Figura seguinte apresenta a curva característica do sensor para cada valor de corrente analisado e seus respectivos valores de tensão de saída (tensão de alimentação: *Tensão de Coletor Comum - Voltage Common Collector (VCC) = 5V*).

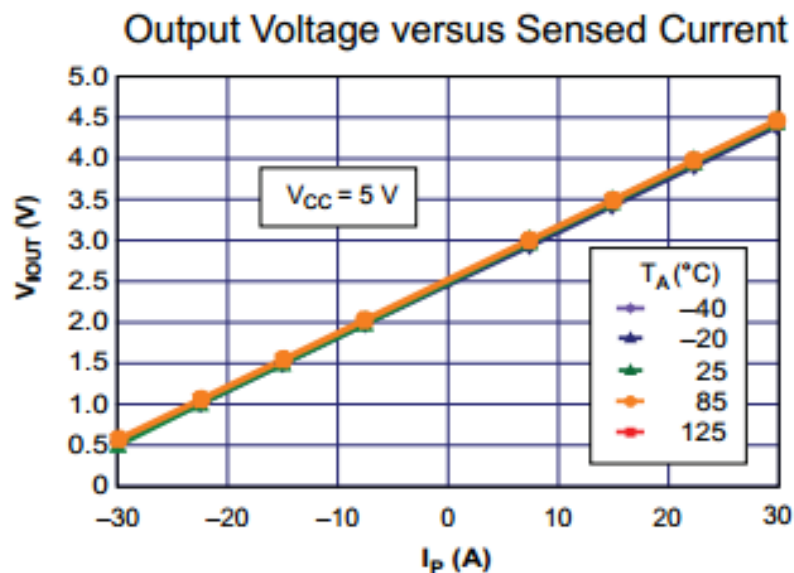


Figura 2.15: Curva característica do sensor de corrente ACS712.

Fonte: <https://www.allegromicro.com//datasheets/acs712-datasheet>

2.3.2 Sensor de Tensão

O Sensor de Tensão AC ZMPT101B é um módulo de alta precisão que tem como finalidade detectar a existência de tensão alternada em um circuito ou fazer a medição do valor de tensão. Este módulo normalmente inclui um transformador de tensão de alta precisão e um circuito amplificador operacional de precisão, tornando-o adequado para medições precisas e confiáveis de tensão CA. Ele pode ser usado para monitorar tensões de rede CA para garantir que os níveis de tensão estejam dentro das faixas aceitáveis para diversas aplicações, como dispositivos de monitoramento de energia e relés de proteção. Contando com apenas quatro pinos de conexão para programação (V_{CC} , saída analógica e dois pinos *Terra - Ground* (GND)) e dois pinos para análise de tensão, o Módulo Sensor de Tensão apresenta uma conexão extremamente simples.

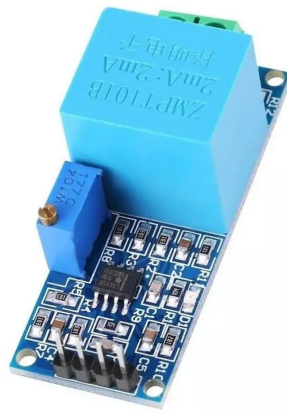


Figura 2.16: Sensor de Tensão AC ZMPT101B. Fonte: <https://www.casadarobotica.com/sensores-modulos/sensores/tensao/sensor-de-tensao-ac-0-a-250v-zmpt101b>

Especificações:

- Tensão de alimentação do módulo: 5 a 30 Volts de Corrente Contínua - Volts Direct Current (VDC);
- Tensão de entrada: 0 a 250 Volts de Corrente Alternada - Volts Alternating Current (VAC);
- Corrente de entrada nominal: 2 mili-Ampère Volts Alternating Current (mA);
- Corrente de saída nominal: 2mA;
- Temperatura de operação: -40° a 70° celsius.

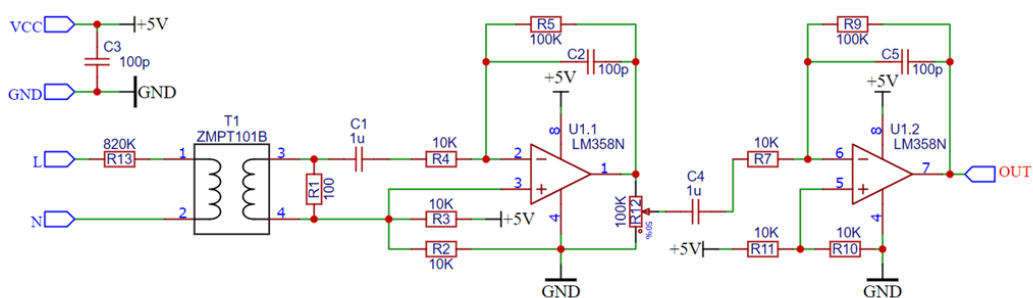


Figura 2.17: Esquema eléctrico do sensor de tensão AC ZMPT101B. Fonte: <https://www.fernandok.com/2022/04/medidor-de-tensao-e-corrente-ac-com.html>

2.3.3 Sensor de presença

Os sensores de presença utilizados em residências são dispositivos electrónicos projetados para identificar o movimento de corpos dentro de uma determinada área de cobertura. Eles funcionam com a tecnologia *Infravermelho Passivo - Passive infrared (PIR)*, que detecta a radiação infravermelha emitida por corpos que emitem calor, como o corpo humano. Sensores PIR permitem detectar movimento, sendo quase sempre usados para identificar se uma pessoa entrou ou saiu da área de alcance do sensor. Eles são pequenos, baratos, consomem pouca energia, são fáceis de usar e não se desgastam com o tempo.

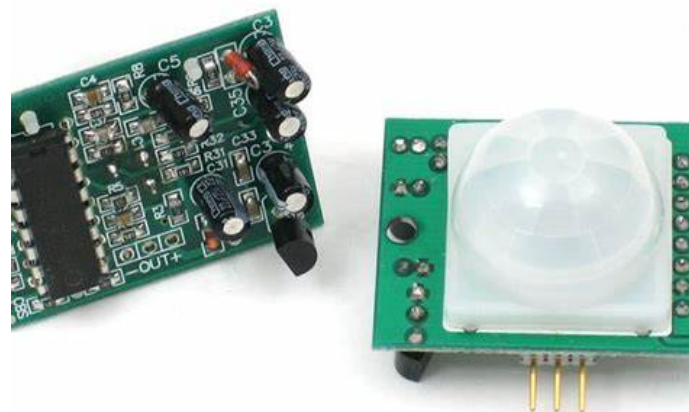


Figura 2.18: Sensor PIR. Fonte: <https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf>.

O sensor em um detector de movimento é, na verdade, dividido em duas metades. Isso acontece porque queremos detectar movimento (mudança), e não níveis médios de radiação infravermelha. As duas metades são conectadas de forma que se anulam mutuamente. Se uma metade detectar mais ou menos radiação infravermelha do que a outra, a saída oscilará para cima ou para baixo.

Quando o sensor está inactivo, ambas as aberturas detectam a mesma quantidade de infravermelho, que é a quantidade ambiente irradiada pelo cômodo, pelas paredes ou do ambiente externo. Quando um corpo quente, como um humano ou animal, passa por perto, ele primeiro intercepta uma das metades do sensor PIR, o que causa uma mudança diferencial positiva entre as duas metades. Quando o corpo quente sai da área de detecção, ocorre o oposto, e o sensor gera uma mudança diferencial negativa. Esses

pulsos de mudança são os que o sensor detecta.

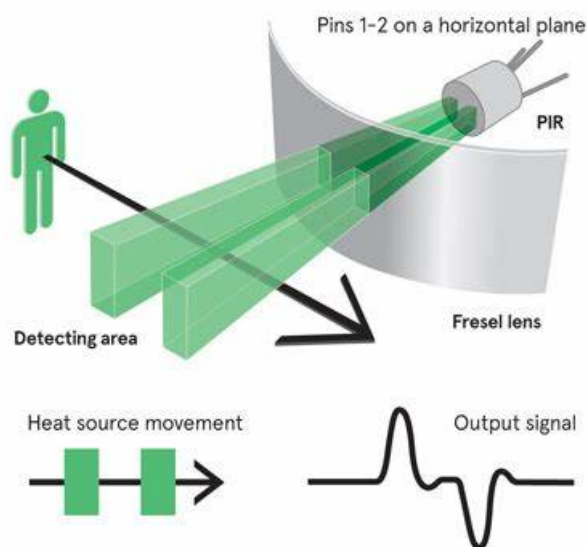


Figura 2.19: Processo de funcionamento do sensor pir. Fonte: <https://forumautomation.com/t/motion-detection-using-pir-passive-infrared-sensor/5485>

A maioria dos módulos PIR possui uma conexão de 3 pinos na lateral ou na parte inferior. Um pino será a terra (GND), outro será o sinal e o último será a alimentação. Este dispositivo possui tensão de operação de 5 a 20 volts, sinal de saída binário (0/ 3,3 volts) e um delay ajustável de 0,3 a 5 minutos. Possui um raio de alcance de 120° até 7 metros (ajustável) e funciona de -15 até 70°C.

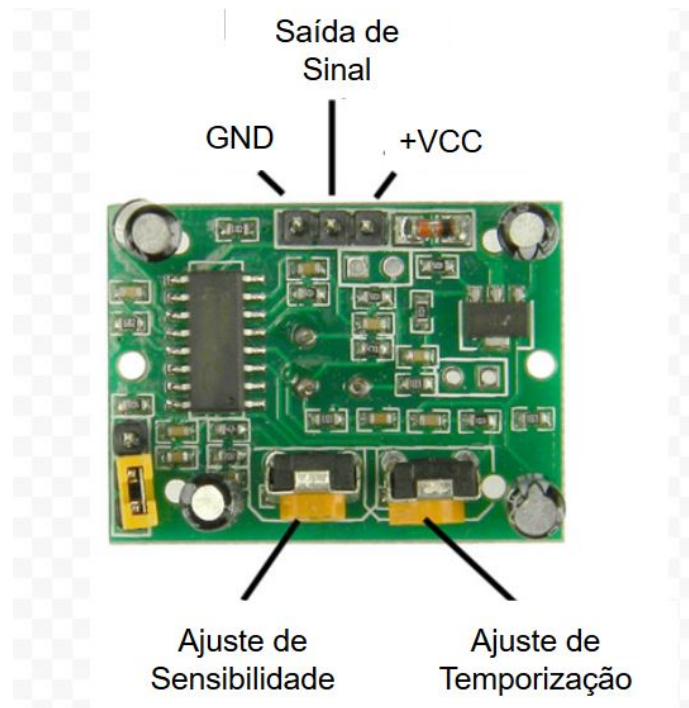


Figura 2.20: Pinagem do sensor pir. Fonte: <https://www.bosontreinamentos.com.br/eletronica/como-funciona-um-sensor-de-movimento-pir-passive-infrared/>

2.3.4 Sensor de temperatura

O sensor de *Umidade e temperatura digitais 22 - Digital humidity and temperature 22* (DHT22) é um dispositivo multifuncional utilizado para medir temperatura e umidade, operando em faixas amplas: de -40 a +80 °C para temperatura e de 0 a 100% de umidade relativa. Ele apresenta uma resposta rápida, com tempo de 2 segundos, e alta precisão, com margem de erro de $\pm 0,5$ °C. Funciona com tensão entre 3V e 5V. Sua saída digital é previamente calibrada, aproveitando uma tecnologia avançada de detecção de umidade e coleta de dados digitais, o que fornece resultados confiáveis e estáveis. Com design compacto, baixo consumo de energia e capacidade de comunicação a distâncias de até 20 metros, é ideal para uso em condições adversas. O encapsulamento em linha única com quatro pinos facilita sua instalação em diversos tipos de projetos eletrônicos.



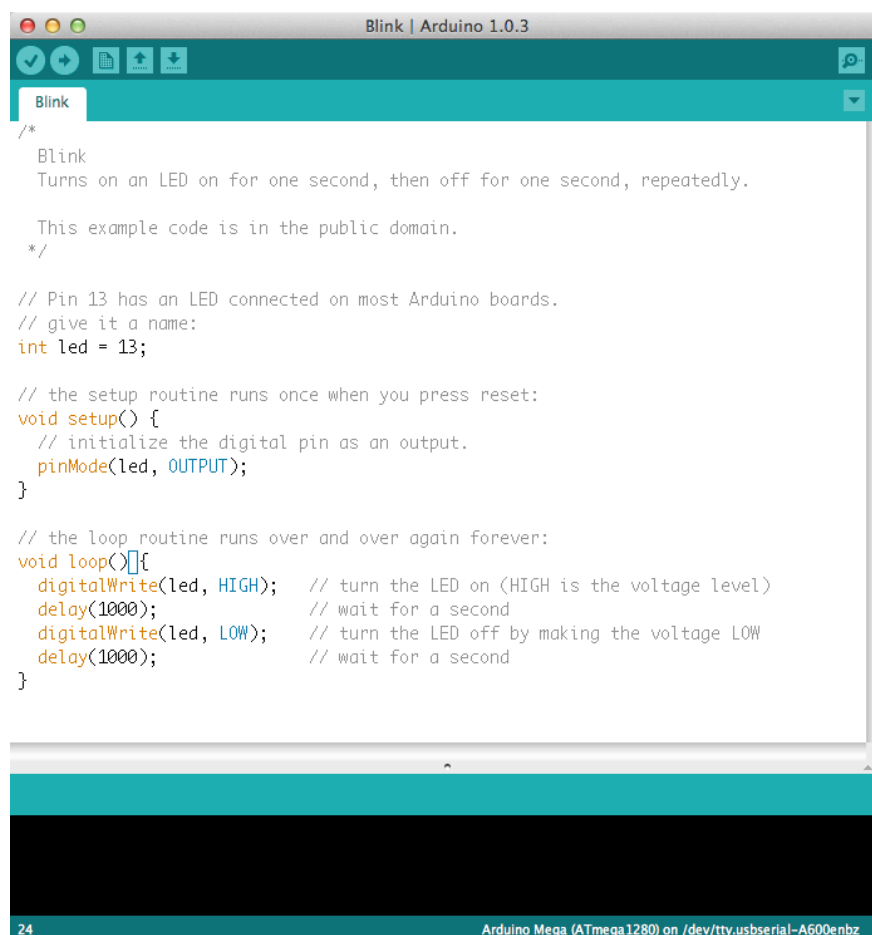
Figura 2.21: Sensor de temperatura. Fonte: <https://www.alldatasheet.com/html-pdf/ETC2/DHT22.html>

O sensor opera com base em dois componentes principais: um termistor, responsável por medir a temperatura e um elemento capacitivo de polímero, que detecta a humidade relativa do ar. Ao ser energizado, o sensor aquece o termistor para captar a temperatura do ambiente. Ao mesmo tempo, o componente capacitivo interage com o vapor d'água presente no ar, absorvendo ou liberando moléculas, ou que altera sua capacitância conforme a umidade. Essas variações são transformadas em sinais digitais que são enviadas ao microcontrolador por meio do pino de dados. O microcontrolador interpreta esses sinais e calcula os valores de temperatura e humidade para exibição.

2.4 Microcontroladores

2.4.1 Plataforma Arduino

O Arduino é uma plataforma utilizada para o desenvolvimento de projetos electrónicos. Ela é composta por uma placa de circuito programável também conhecida como microcontrolador, e por um software, chamado *Ambiente de Desenvolvimento Integrado - Integrated Development Environment (IDE)*, que é instalado no computador e serve para escrever e transferir códigos para a placa. Com o Arduino, é possível se comunicar com diversos outros dispositivos electrónicos, enviando e recebendo dados. Um destaque importante é que todo o seu conteúdo – incluindo software, bibliotecas e hardware – é de código aberto, ou seja, pode ser utilizado livremente por qualquer pessoa, sem custos com licenças ou direitos autorais. Estes factores contribuíram para a integração da plataforma no projecto.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0.3". The main editor area contains the following code:

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop(){
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

The status bar at the bottom indicates "24" on the left and "Arduino Mega (ATmega1280) on /dev/tty.usbserial-A600enbz" on the right.

Figura 2.22: IDE do arduino. Fonte:<https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>

O projeto faz uso da placa Arduino Uno R3(no subsistema de controle), que é uma plataforma de prototipagem baseada no microcontrolador ATmega328P. Essa placa conta com 14 pinos digitais de entrada e saída (sendo 6 deles compatíveis com saída *Modulação por Largura de Pulso - Pulse Width Modulation (PWM)*), além de 6 entradas analógicas. Ela opera com um ressonador cerâmico de 16 *Mega-Hertz (MHz)*, responsável por gerar o relógio que determina a velocidade de execução das instruções. Também possui porta *Barramento serial universal - Universal Serial Bus (USB)* para comunicação com o computador, conector de alimentação externa, interface *Programação Serial em Circuito - In-Circuit Serial Programming (ICSP)* para programação e um botão de reinicialização. A tensão de funcionamento é de 5V, e a placa dispõe de 32 *Kilo-Byte (KB)* de memória Flash para o código do programa, 2 kB de *Memória Estática de Acesso Aleatório - Static Random Access Memory (SRAM)* e 1 kB de EEPROM para armazenamento de dados.

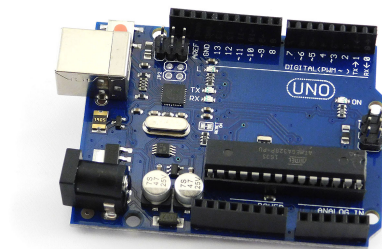


Figura 2.23: Arduino Uno R3. Fonte:<https://nettigo.eu/products/uno-r3-clone-atmega328p-mega16u2>

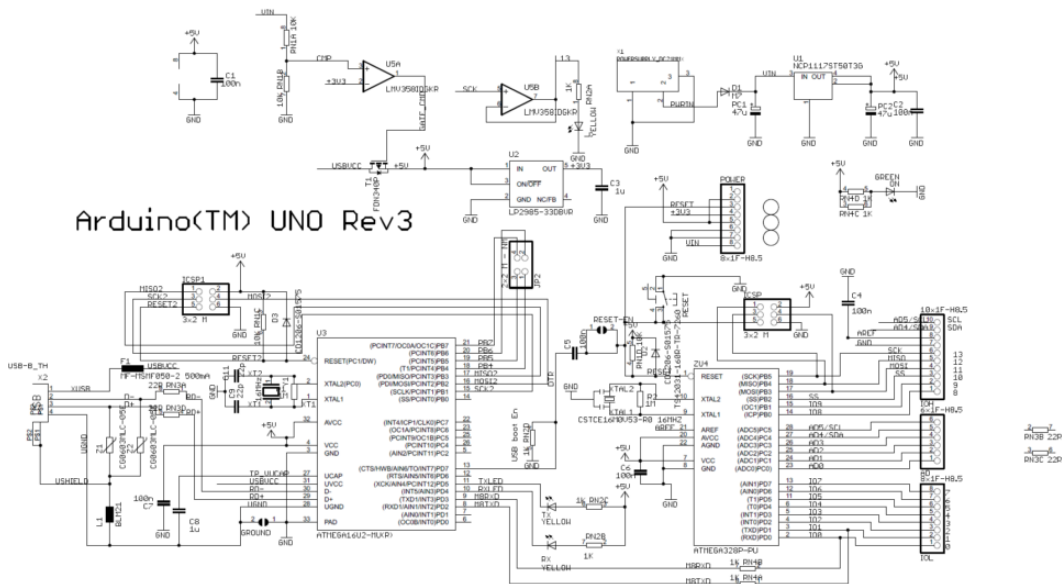


Figura 2.24: Esquema eléctrico do arduino uno R3.

Fonte: <https://docs.arduino.cc/hardware/uno-rev3/tech-specs>.

2.4.2 ESP32

O *Espressif Systems Platform 32 (ESP32)* é um microcontrolador e *Sistema em um Chip - System on a Chip (SoC)* desenvolvido pela Espressif Systems, amplamente utilizado em aplicações de *Internet das coisas - Internet of Things (IoT)* e sistemas embarcados. Ele possui um processador Xtensa LX6 dual-core, com cada núcleo operando a até 240 MHz, oferecendo bom desempenho e capacidade multitarefa. Com 520 KB de SRAM e 4 *Mega-Byte (MB)* de memória flash, fornece espaço suficiente para código e dados. O ESP32 conta com conectividade *Wireless Fidelity (WiFi)* (802.11 b/g/n) e Bluetooth (*Bluetooth de Baixo Consumo de Energia - Bluetooth Low Energy (BLE)* e Classic), além de suportar modos cliente e ponto de acesso. Dispõe de diversos pinos *Entrada/Saída de Propósito Geral - General Purpose Input/Output (GPIO)* para interface com sensores e dispositivos externos, *Conversor Analógico-Digital com Registro de Aproximação Sucessiva - Analog-to-Digital Converter – Successive Approximation Register (ADC-SAR)* de 12 bits com múltiplos canais, dois canais *Conversor Analógico-Digital - Analog-to-Digital Converter (ADC)* de 8 bits e diversos periféricos integrados como *Transmissor/Receptor Assíncrono Universal - Universal Asynchronous Receiver/Transmitter (UART)*, *Circuito Inter-Integrado - Inter-Integrated Circuit (I2C)*, *Interface Serial Periférica - Serial Peripheral Interface (SPI)* e PWM. Alguns módulos incluem sensores embarcados, como de tem-

peratura e umidade, e o chip também oferece modos de baixo consumo de energia, ideais para dispositivos alimentados por bateria. Pode ser programado por meio do Arduino IDE, PlatformIO ou *Espressif IoT Development Framework* (ESP-IDF), e é amplamente aplicado em dispositivos , automação residencial, wearables, módulos de comunicação, eletrodomésticos inteligentes e outras soluções conectadas.

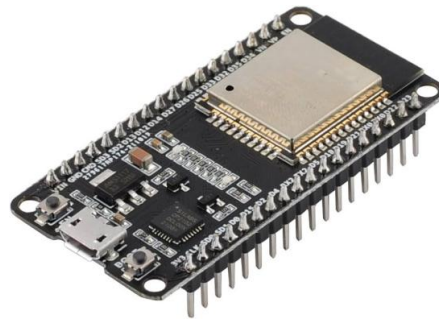


Figura 2.25: ESP32. Fonte: <https://victorvision.com.br/blog/placa-esp32/>

Pinagem do esp32

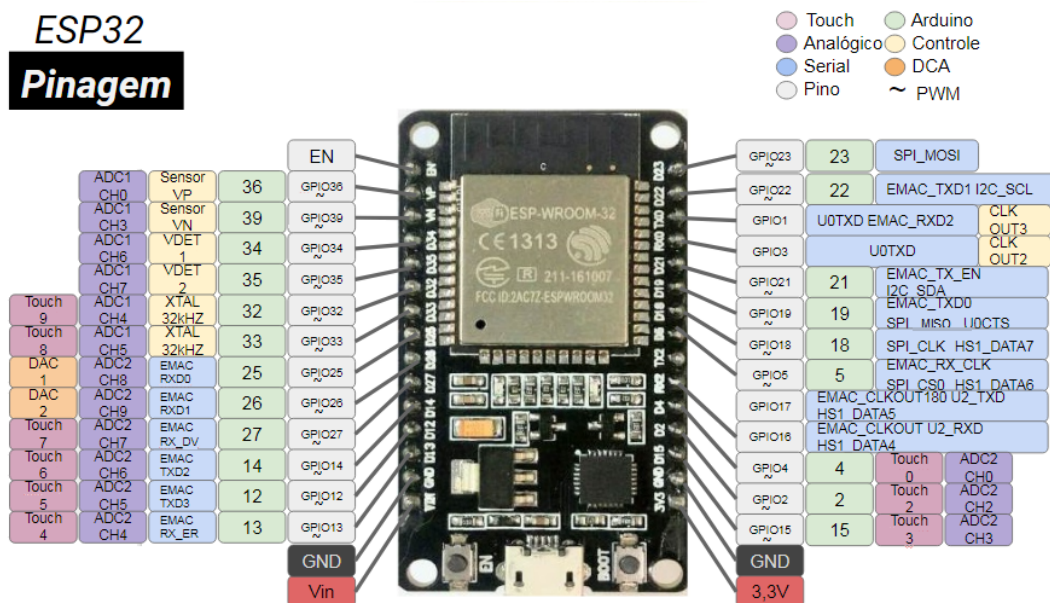


Figura 2.26: Pinagem do esp32. Fonte: <https://curtocircuito.com.br/blog//conhecendo-esp32>

O ESP32 é uma placa de desenvolvimento com 25 GPIOs digitais, sendo que os pinos 34, 35, 36 e 39 são apenas de entrada. Alguns pinos possuem pull-up interno (como os GPIOs 14, 16 a 23), enquanto outros exigem pull-up externo. Ele conta com 15 entradas analógicas (ADC) de 12 bits, permitindo converter tensões de 0 a 3,3V em valores digitais

de 0 a 4095, e dois canais DAC de 8 bits para gerar sinais analógicos. A placa inclui 9 GPIOs capacitivos com detecção de toque, funcionando como botões sensíveis ao toque. Possui duas interfaces I2C com pinos configuráveis, sendo os GPIOs 21 (*Linha de Dados Serial - Serial Data Line (SDA)*) e 22 (*Linha de Clock Serial - Serial Clock Line (SCL)*) os mais usados. Há também três interfaces UART para comunicação serial e 21 canais PWM para controle de motores e LEDs. Além disso, conta com pinos de alimentação, boot, *Relógio em Tempo Real - Real-Time Clock (RTC)*, SPI e pino de enable (EN).

2.5 Actuadores

Atuadores são dispositivos que convertem um tipo de energia em movimento mecânico. Eles transformam sinais eléctricos, hidráulicos ou pneumáticos em ações físicas, como mover um cilindro, girar um motor ou ajustar uma válvula.

Podem ser eléctricos, oferecendo precisão e controle por meio de motores, servos ou motores de passo; pneumáticos, que utilizam ar comprimido para movimentos rápidos e simples; hidráulicos, que usam fluido sob pressão para gerar força elevada; mecânicos, que operam com alavancas, roscas ou engrenagens; e térmicos, que aproveitam a dilatação de materiais com o calor.

2.5.1 Módulo Relé

Módulo relé é um componente electrónico que funciona como um interruptor acionado por corrente eléctrica. Ele é formado por uma bobina e contatos metálicos. Ao aplicar uma tensão na bobina, um campo magnético é gerado, fazendo com que os contatos mudem de posição.

Características principais:

- Funciona com alimentação de 5V;
- Compatível com diferentes tensões na saída;
- Tempo de acionamento entre 5 e 10 milissegundos;
- Capaz de ligar ou desligar cargas de até 220V em corrente alternada.

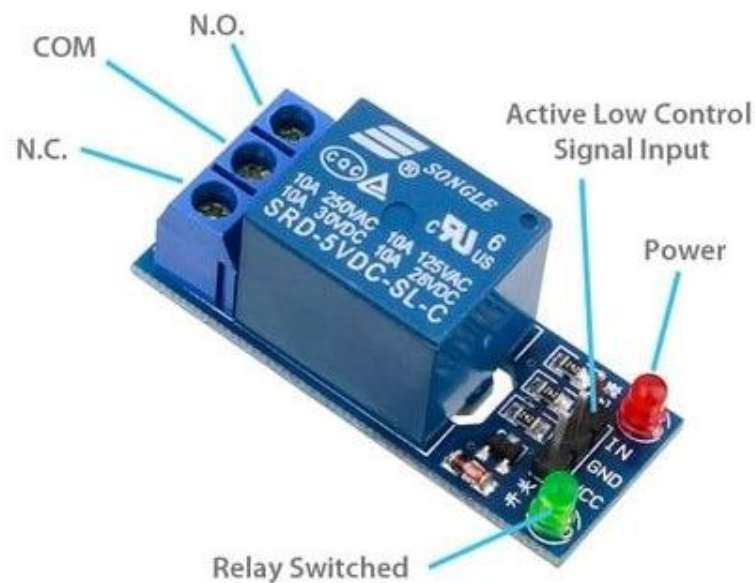


Figura 2.27: Módulo Relé. Fonte: <https://www.mactronica.com.co/modulo-rele-1-canal>

Quando os contatos estão normalmente abertos, eles se fecham com a ativação do relé; se estiverem normalmente fechados, se abram. Isso possibilita o controle de corrente eléctrica em circuitos externos. Os relés são frequentemente usados para permitir que circuitos de baixa potência acionem equipamentos de maior consumo de energia com segurança, como motores, lâmpadas e outros dispositivos.

2.6 Programação

Programação refere-se ao processo de desenvolvimento, escrita, teste e manutenção de programas de computador através de linguagens de programação.

Uma linguagem de programação é uma forma estruturada de comunicação usada para instruir o computador a realizar determinadas tarefas. Ela é formada por dois elementos principais:

- a. Um vocabulário específico, composto por palavras, símbolos e sinais com significados bem definidos (semântica);

- b. Um conjunto de regras que define como esses elementos devem ser combinados para formar comandos válidos (sintaxe).

O principal objetivo das linguagens de programação é permitir a criação de soluções para problemas diversos por meio do uso do computador. Existem muitas linguagens de programação, cada uma com suas próprias particularidades e aplicações. Alguns exemplos são: Python, Java, Pascal, Clipper, C, Visual Basic, Delphi, entre outras.

2.6.1 Tipos de linguagens de programação

As linguagens de programação são classificadas em dois tipos principais: linguagem de baixo e alto nível.

As linguagens de programação de baixo nível são totalmente orientadas à máquina, funcionando como uma interface direta entre o software e o hardware. Elas permitem um controle preciso sobre o equipamento e sua estrutura física, criando um vínculo inseparável entre os dois. Para utilizá-las adequadamente, é fundamental que o programador tenha um conhecimento profundo do hardware, pois essas linguagens exigem uma compreensão detalhada da arquitetura da máquina. Exemplos incluem Assembly e C.

A linguagem de alto nível é aquela cuja sintaxe é voltada para o entendimento humano. Isso a torna mais intuitiva, amigável e fácil de aprender, especialmente para iniciantes na área de programação. Essas linguagens abstraem os conceitos técnicos relacionados ao funcionamento da máquina, permitindo ao programador focar na lógica do problema. Comandos são sintetizados em palavras de fácil compreensão, como “print” para exibir informações na tela ou “delete” para excluir dados, facilitando ainda mais o processo de desenvolvimento. Alguns exemplos incluem: Python, Java e JavaScript.

2.6.2 Linguagem C++

A linguagem C++ foi criada por Bjarne Stroustrup dos Bell Labs na década de 1980 . Seu principal objetivo era implementar uma versão distribuída do núcleo Unix.

C++ é uma linguagem de programação tipada estaticamente e de propósito geral, desenvolvida para oferecer eficiência e portabilidade comparáveis às da linguagem C. Ela foi projetada com suporte a múltiplos paradigmas de programação, como o imperativo, orientado a objetos e genérico, proporcionando ao programador diversas escolhas, ainda que essas escolhas possam incluir opções propensas a erros. Um dos princípios funda-

mentais do C++ é a compatibilidade com C, permitindo uma transição suave e o reaproveitamento de código legado. A linguagem evita incluir recursos que sejam específicos de determinadas plataformas ou voltados apenas para grupos restritos de desenvolvedores, buscando manter sua ampla aplicabilidade. Além disso, o C++ é projetado para não impor custos de execução (overhead) para funcionalidades que não são utilizadas, prezando pela eficiência. Por fim, ele pode ser utilizado mesmo em ambientes com recursos limitados, sem a exigência de ferramentas sofisticadas de desenvolvimento, o que reforça sua flexibilidade e abrangência.

A linguagem C++ deriva da C e mantém uma sintaxe semelhante, oferece suporte à programação orientada a objetos. Além disso, C++ é uma linguagem portátil, ou seja, pode ser compilada e executada em diferentes sistemas operacionais.

2.7 Web

A Web, ou World Wide Web (Teia de Alcance Mundial), é o nome dado à rede de servidores na Internet que armazena documentos hipermídia conectados uns aos outros por meio de hiperlinks, criando uma vasta rede de informações distribuídas globalmente. Criada por Tim Berners-Lee em 1989, ela revolucionou a maneira como as informações são compartilhadas e consumidas, tornando possível a comunicação instantânea e o acesso a uma imensa variedade de dados em qualquer lugar do mundo.

Ela funciona por meio de um conjunto de protocolos de comunicação, sendo o HTTP (Hypertext Transfer Protocol) o principal deles. Quando um usuário digita o endereço de um site em seu navegador, uma solicitação é enviada a um servidor responsável por hospedar aquele conteúdo. O servidor, então, processa essa solicitação e envia os dados correspondentes de volta ao navegador, que os interpreta e exibe na tela, permitindo a interação com o conteúdo de forma dinâmica e acessível.

Os principais componentes da Web incluem os navegadores, os servidores e as URLs (Uniform Resource Locators). Os navegadores são softwares responsáveis por interpretar e exibir as páginas da Web de forma acessível ao usuário. Já os servidores armazenam e disponibilizam os conteúdos solicitados pelos navegadores. As URLs, por sua vez, funcionam como endereços únicos que identificam a localização de recursos na Web, permitindo que os usuários acessem as informações desejadas de forma rápida e precisa.

2.7.1 Programação web

Programação web envolve a criação de páginas e aplicações web usando várias tecnologias. Os conceitos fundamentais abrangem linguagens de marcação, estilo e programação. As linguagens principais incluem HTML (HyperText Markup Language), XML (Extensible Markup Language), CSS (Cascading Style Sheets) e JavaScript.

2.7.1.1 HTML (HyperText Markup Language)

O HTML é a base da construção de páginas na internet, servindo para organizar e estruturar os elementos presentes em um site. Ele utiliza marcas (tags) que delimitam e atribuem significado ao conteúdo. Com HTML, é possível disponibilizar textos, imagens, tabelas e diversos outros tipos de conteúdo online; acessar informações por meio de links clicáveis; criar formulários para realizar transações comerciais, como buscas, reservas e pedidos; além de incorporar recursos multimídia como áudio, vídeo e aplicações interativas.

2.7.1.2 CSS (Cascading Style Sheets)

CSS serve para definir a aparência visual do conteúdo em HTML, controlando como as páginas web são exibidas. Com ele, é possível ajustar cores, tipografias, espaçamentos, organização dos elementos e outros detalhes visuais. Isso facilita a separação entre estrutura e estilo, tornando o código mais fácil de manter e garantindo uniformidade no design.

2.7.1.3 JavaScript

JavaScript é uma linguagem de programação avançada que roda diretamente no navegador do usuário, permitindo tornar as páginas web mais interativas e dinâmicas. Ele possibilita alterar o conteúdo HTML, reagir a ações do usuário, validar formulários e executar diversas outras funções.

Capítulo 3

Desenho & Implementação

Neste capítulo, são apresentados os componentes e os circuitos utilizados na implementação do hardware do sistema, assim como as ferramentas empregadas no desenvolvimento do software. Além disso, explora-se o desenvolvimento do sistema como um todo, explicando o papel e o funcionamento de cada subsistema, bem como a contribuição dos diferentes componentes para o desempenho global. Também são descritas as interligações entre os subsistemas e as etapas envolvidas na criação do software que integra o sistema.

3.1 Dimensionamento do sistema

3.1.1 Subsistema de medição do consumo de energia

3.1.1.1 Esquema eléctrico

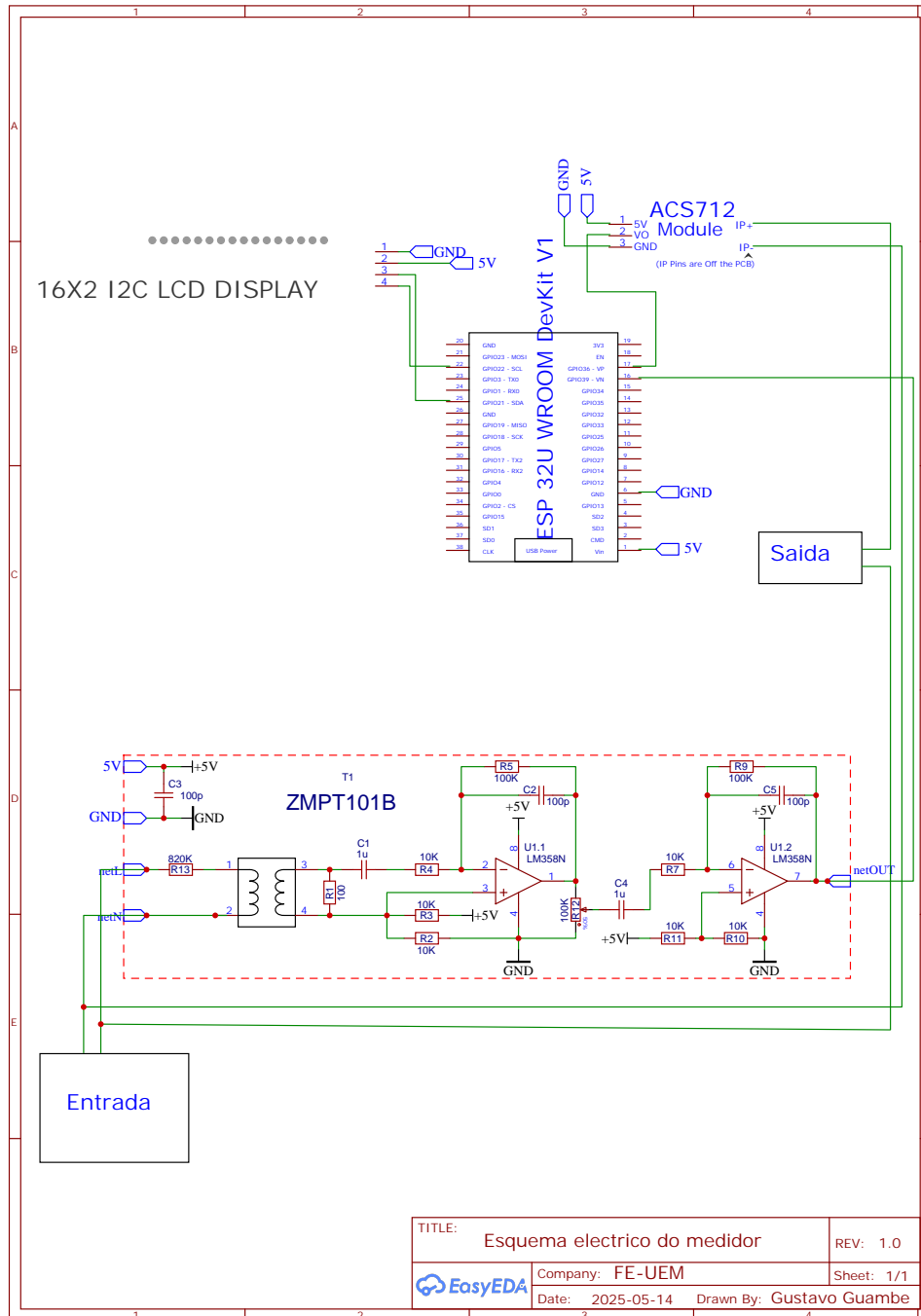


Figura 3.1: Esquema eléctrico do medidor. Fonte: Autor

Os pinos de saída dos sensores de corrente e tensão foram ligados aos pinos GPIO's 36 e 32 do ESP32 respectivamente. O módulo I2C do display LCD conecta-se com o ESP32

por meio dos pinos SDA e SCL que são comumente utilizados como pinos I2C padrão. Os sensores de corrente e tensão, o display LCD com módulo I2C e o ESP32 operam todos com a mesma tensão. Por esse motivo, seus pinos de alimentação foram conectados em paralelo e ligados a uma fonte de alimentação de 5V. Os terminais de conexão do sensor de corrente foram conectados de tal forma que o sensor esteja em série com a carga que se pretende medir, ele foi montado na linha de fase tendo se interrompido o circuito para fazer a conexão, já que se trata de um sensor invasivo.

3.1.1.2 Componentes

Para a materialização do subsistema de medição do consumo de energia, são necessários sensores de corrente e tensão, além de um microcontrolador. Também é necessária uma interface para a visualização dos dados de consumo.

Tabela 3.1: Componentes do subsistema de medição. Fonte: Autor

Componente	Descrição
Sensor de Corrente ACS712	Este sensor mede corrente eléctrica e foi selecionado para o projecto por ser de baixo custo. Ele é capaz de medir correntes de até 30 A, sendo ideal para uso residencial.
Sensor de Tensão ZMPT101B	Este sensor mede tensão eléctrica e foi selecionado para o projecto devido à sua alta precisão. Ele é capaz de medir tensões de até 250 VCA, o que o torna adequado para aplicações em redes eléctricas residenciais que operam a 220 VCA.
Microcontrolador ESP32	Este microcontrolador foi escolhido principalmente pela sua capacidade de operar como servidor e hospedar páginas web. Um dos seus principais diferenciais é a integração nativa com Wi-Fi e Bluetooth, o que o torna especialmente versátil para aplicações conectadas.
Display LCD 16x02	Interface de visualização dos dados de consumo. O dispositivo foi selecionado devido ao seu baixo custo, tornando-se uma opção economicamente viável para o projecto.
Módulo I2C	Facilita a integração e o controle do display LCD, simplificando a comunicação e reduzindo a quantidade de fios necessários.

3.1.2 Subsistema de controle do consumo de energia

3.1.2.1 Esquema eléctrico

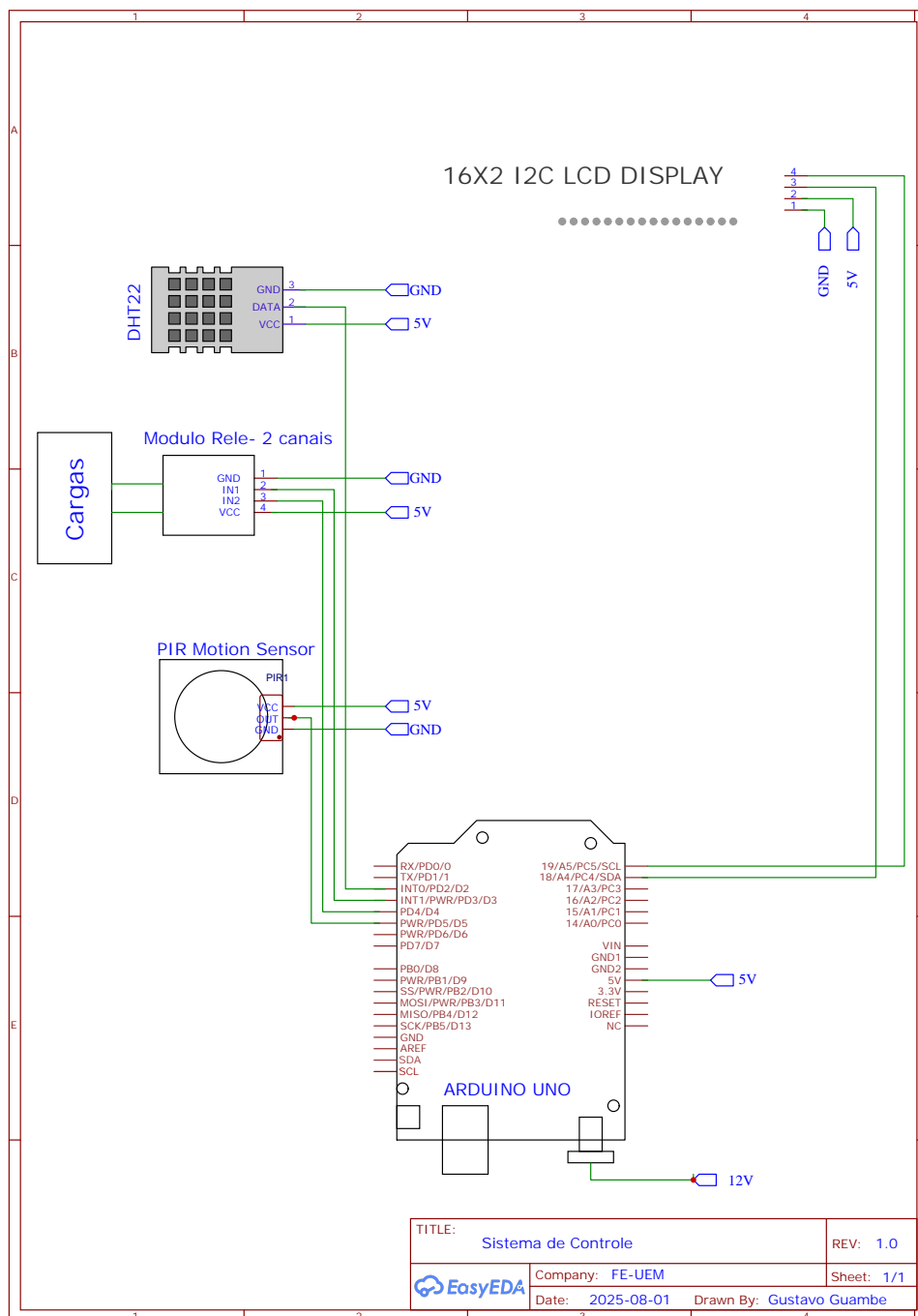


Figura 3.2: Esquema eléctrico do sistema de controle. Fonte: Autor

O pino de dados do sensor de temperatura DHT22 foi conectado à porta digital 2 do Arduino, enquanto o pino de saída do sensor PIR foi ligado à porta digital 5. As entradas digitais IN1 e IN2, responsáveis por acionar os relés, foram conectadas às portas digitais 3 e 4 do Arduino, respectivamente. O módulo I2C do display LCD conecta-se com o arduino

por meio dos pinos SDA e SCL que são comumente utilizados como pinos I2C padrão. Os sensores de temperatura DHT22 e de presença PIR, o módulo relé e o display LCD junto com o módulo I2C são energizados pelo arduino ligando os seu pinos de alimentação em paralelo e conectando-os aos pinos 5V e GND do arduino. O arduino por sua vez é energizado usando uma fonte de alimentação externa de 12V.

No que concerne às ligações externas o circuito do ventilador opera com uma tensão de 12V, a mesma utilizada para energizar o arduino sendo por isso alimentado através do pino Vin do arduino. O circuito da lâmpada opera com 220V. As duas cargas foram conectadas nas portas NA dos seus relés de modo que enquanto os relés não estiverem acionados as cargas permanecem desligadas.

3.1.2.2 Componentes

Para a implementação do subsistema de controle são necessários sensores para a captação de dados, relés para o acionamento de cargas e um microcontrolador para o processamento das informações. Além disso, é essencial uma interface para a visualização dos dados coletados.

Tabela 3.2: Componentes do subsistema de controle. Fonte: Autor

Componente	Descrição
Sensor de temperatura DHT22	É responsável pela medição da temperatura. Foi escolhido para o projecto por oferecer maior precisão em comparação com outros sensores similares.
Sensor Presença Pir	Detecta movimento. Foi selecionado devido ao seu baixo custo.
Microcontrolador Arduino Uno	É responsável pelo processamento dos dados e envio de sinais de controle. Foi selecionado devido ao seu baixo custo.
Módulo Relé de 2 canais	É responsável por ligar/desligar os equipamentos.
Display LCD 16x02	Interface de visualização dos dados. O dispositivo foi selecionado devido ao seu baixo custo, tornando-se uma opção economicamente viável para o projecto.
Módulo I2C	Facilita a integração e o controle do display LCD, simplificando a comunicação e reduzindo a quantidade de fios necessários.

3.1.3 Linguagem de programação usada

O código-fonte dos dois subsistemas — o sistema de medição e o sistema de controle — foi desenvolvido na IDE do Arduino, utilizando a linguagem C++. Nesse ambiente, o C++ é aplicado de forma simplificada, o que facilita a programação de microcontroladores e torna o desenvolvimento mais acessível.

3.1.4 Página web

O desenvolvimento do front-end foi realizado com o uso das seguintes tecnologias:

- HTML, responsável pela estruturação do conteúdo da interface;
- CSS, utilizado na definição do estilo e dos elementos visuais;
- JavaScript, empregado para implementar a interactividade e permitir a comunicação com o usuário.

O back-end foi desenvolvido em C++ utilizando a IDE do Arduino e está integrado diretamente ao código-fonte do subsistema medidor. Essa integração permite a aquisição e o processamento eficiente dos dados, que são transmitidos em tempo real para a página, garantindo uma comunicação contínua e sincronizada entre o sistema de medição e a interface.

3.2 Implementação do sistema

3.2.1 Descrição do sistema proposto

O sistema proposto visa medir e controlar o consumo de energia eléctrica de um ambiente com dois cômodos e enviar notificações ao usuário acerca desse consumo. Será usado um equipamento específico em cada cômodo, um ventilador e uma lâmpada. O sistema contará com um subsistema de medição que vai medir os valores de corrente e tensão eléctrica e, a partir desses dados, calcular a potência activa e o consumo de energia nos dois cômodos. Um subsistema de controle será responsável por activar e desactivar os equipamentos através de relés, com o intuito de evitar o desperdício de energia. O sistema também contará com uma interface que permitirá ao consumidor acessar remotamente os dados de consumo e receber notificações.

3.2.1.1 Diagrama de blocos

A figura seguinte ilustra o diagrama de blocos do sistema.

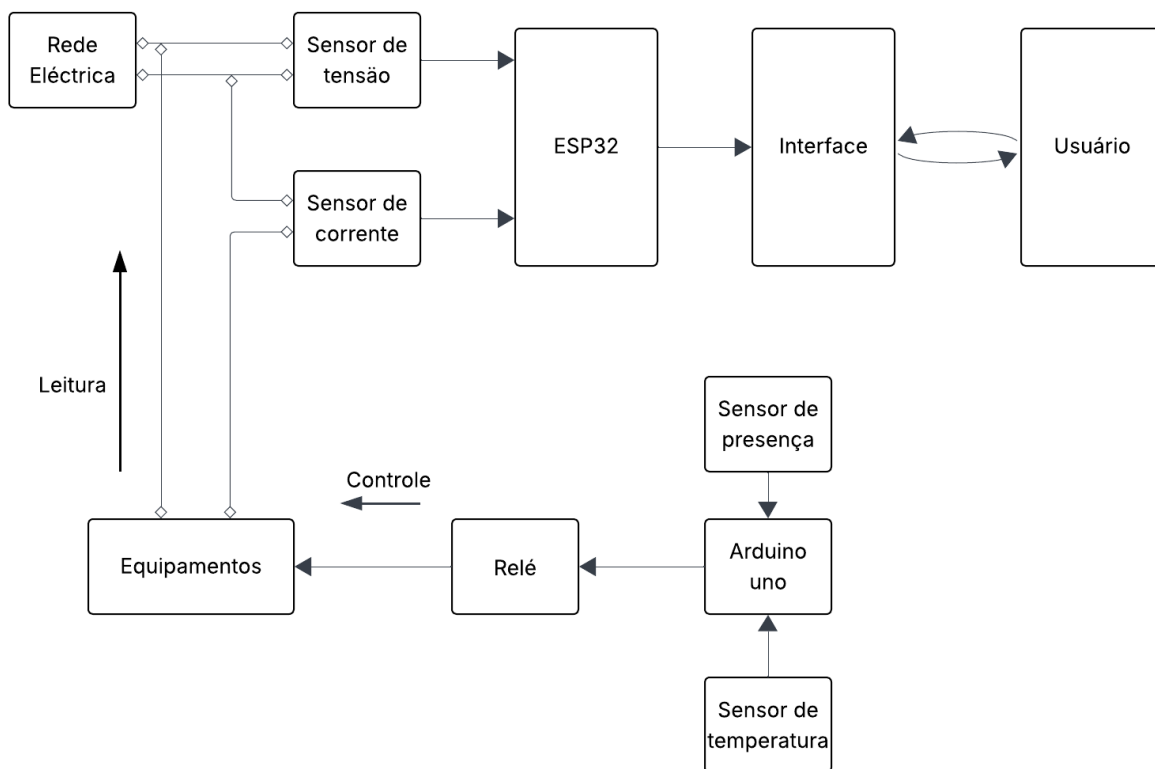


Figura 3.3: Diagrama de blocos do sistema. Fonte: Autor

3.2.2 Subsistema de medição do consumo de energia

O Subsistema de medição é controlado pelo ESP32 e conectado à rede eléctrica através dos sensores de tensão e corrente. O sensor de tensão mede a tensão da rede eléctrica enquanto o sensor de corrente registra a corrente consumida pelos cômodos. Esses dados são recebidos pelo ESP32, que realiza os cálculos da potência activa e do consumo de energia, e depois envia os dados para um display LCD e para uma página na web.

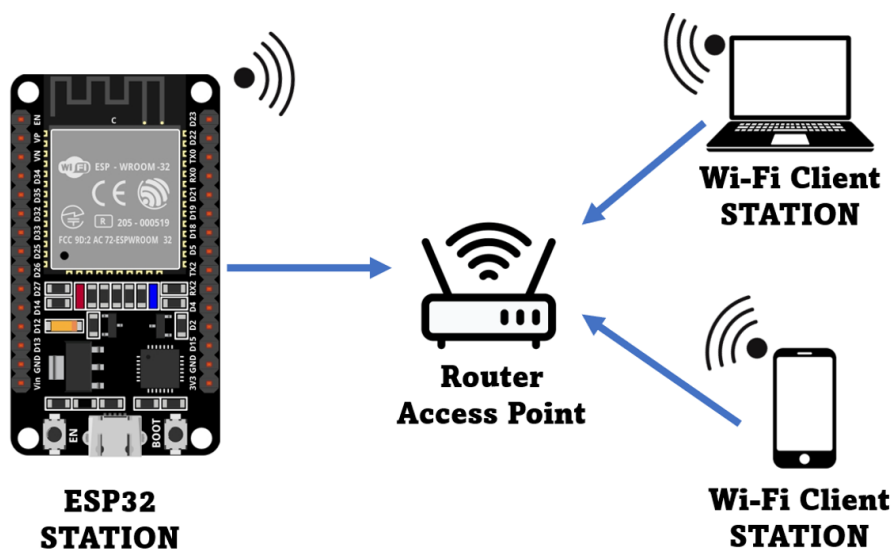


Figura 3.4: Servidor web ESP32 como uma estação WIFI. Fonte: <https://linuxhint.com/esp32-wifi-arduino-ide/>

O ESP32 possui Wi-Fi e Bluetooth integrados, neste projecto ele foi programado para fazer uma conexão WI-FI, ou seja, o subsistema de medição só fará medições se o ESP32 estiver conectado a uma rede wifi(roteador). Os dados desta rede (nome e senha) devem estar inseridos no código que será carregado no ESP32 . Quando o ESP32 estiver conectado ao Wi-Fi,as medições serão iniciadas.

O cálculo do consumo de energia é apresentado a seguir, baseado na Equação 2.13 do capítulo 2.

$$Consumo(Wh) = \frac{\sum P_i}{3600} \quad (3.1)$$

Esta fórmula usa as medições de potência activa feitas a cada segundo. Essas potências activas medidas em watts são somadas o que equivale a somar a energia instantânea ao longo do tempo. Para transformar essa soma de potências instantâneas de watts·segundo para watt-hora é feita a divisão por 3600(número de segundos em uma hora).

3.2.3 Subsistema de controle do consumo de energia

O subsistema de controle é constituído por um Arduíno uno, relés e sensores de captação de dados. Os sensores realizam a leitura das condições dos cômodos e transmitem essas informações para o arduino. Com base nesses dados, o Arduíno realiza os cálculos e operações necessários para realizar o controle dos equipamentos. Em seguida, sinais de

controle são enviados para os relés para activar ou desactivar os equipamentos. Além disso, o subsistema conta com uma interface que exhibe a temperatura e o estado dos cômodos onde os sensores estão instalados.

3.2.4 Código fonte

3.2.4.1 Código fonte do subsistema de medição do consumo de energia

A medição da tensão e da corrente eléctrica, assim como o cálculo da potência activa, foi realizada por meio da biblioteca *EmonLib.h*, utilizada no ESP32 e fornecida pelo site *OpenEnergyMonitor*. Essa biblioteca é responsável por interpretar os sinais analógicos enviados pelos sensores e, de forma automática, calcular os valores eficazes da tensão da rede e da corrente consumida, além da potência activa (real), da potência aparente e do fator de potência, entregando esses dados já processados.

O código inclui também as bibliotecas *WiFi.h*, *WebServer.h*, *WebSocketsServer.h* e *LiquidCrystalI2C.h*.

- *WiFi.h* permite trabalhar com a placa de rede do nosso dispositivo;
- *WebServer.h* para conexão entre cliente navegador web e servidor esp32. Possui as funções para hospedar a página web;
- *WebSocketsServer.h* mantém uma conexão contínua entre o servidor esp32 e cliente navegador web de modo a transmitir os dados em tempo real para a página;
- *LiquidCrystalI2C.h* é usada para controlar o módulo I2C, que facilita a utilização do display LCD no ESP.

Depois de incluir as bibliotecas são colocadas as credenciais de acesso nomeadamente o nome da rede (*Identificador do Conjunto de Serviço - Service Set Identifier (SSID)*) e a respectiva senha a que o módulo ESP32 estará conectado. É feita a criação de um objecto web Server na porta 80 de modo idêntico é criada uma instância web Socket na porta 81. De seguida é declarada uma variável relacionada ao funcionamento do código: A variável *JSONtxt* do tipo String que será usada para enviar os valores de corrente, tensão e potência para a página web via WebSocket em formato *Notação de Objeto JavaScript - JavaScript Object Notation (JSON)*. Antes da função *setup*, é feita a declaração da função *handleRoot()* que define o *Rede Mundial de Computadores - World Wide Web (WEB) Server* e tem como objectivo exhibir a interface de conexão para o usuário. Temos também

a função `WebSocketEvent` que aciona os eventos quando uma solicitação é recebida. A segunda parte do código é destinada à função `setup`, estrutura básica na programação do Arduino. A serial é iniciada com uma taxa de 9600 bauds. Nessa parte encontra-se a inicialização e as primeiras escritas no Display LCD e a configuração da conexão wifi. Ainda na mesma estrutura, a função `emon1.voltage()` é responsável por definir os parâmetros da leitura de tensão da instância `emon1`. Similarmente, a função `emon1.current()` recebe os parâmetros de medição de corrente da mesma instância.

A terceira parte do código é dedicada à função `loop`, também fundamental e responsável por criar uma recursão permanente. A função `emon1.calcVI()` é responsável por realizar os cálculos de tensão e corrente *Raiz Média Quadrática - Root Mean Square (RMS)* da instancia `emon1` e, em seguida, potência. Os cálculos dos valores realizados pela biblioteca são apresentados a seguir, baseados em carregamentos não-lineares.

Cálculo da tensão RMS:

Os valores instantâneos de tensão são multiplicados entre si um número de vezes igual ao número de amostras interno da biblioteca e o resultado destes são adicionados a um somatório. A média dos valores quadrados é obtida pela razão entre o somatório e o números de amostras, com a raiz deles sendo aplicada nesta média.

```
emon1.calcVI(20, 100); // 20-nr de meio ciclos, 100- tempo limite
```

```
float supplyVoltage = emon1.Vrms;
```

Cálculo da corrente RMS:

Similarmente como é feito na tensão RMS, a raiz da média dos valores quadrados da corrente é obtida, através dos valores instantâneos de corrente.

```
double Irms = emon1.calcIrms(1480); // 1480-indica o nr de amostras usadas no algoritmo
```

Cálculo da potência:

Os valores instantâneos de corrente e tensão medidos são multiplicados entre si, obtendo-se o valor de potência instantânea. Este é somado um número de vezes igual ao número de amostras interno da biblioteca e esta soma total, dividida por esta quantidade de amostras, resultando na média dos valores de amostras.

```
float pot = (supplyVoltage * Irms) / 1000;
```

ESCRITA NO DISPLAY

```
lcd.setCursor(0, 0);
```

```
lcd.print("I:" + String(Irms, 1) + "A");
```

```
lcd.setCursor(9, 0); lcd.print("U:" + String(supplyVoltage) + "V"); lcd.setCursor(0, 1);
```

```
lcd.print("Pot:" + String(pot, 1) + "KW");
```

Envio dos dados para a página web

Os valores de corrente, tensão e potência são enviados para o soquete e quaisquer alterações nesses valores são refletidas imediatamente na página web.

```
String corr = String(Irms);  
String tens = String(supplyVoltage);  
String potenc = String(pot);  
JSONtxt = TEMP + corr + ",";  
JSONtxt += HUM + tens + ",";  
JSONtxt += POT + potenc + "`"  
webSocket.broadcastTXT(JSONtxt);  
Serial.println(JSONtxt);
```

3.2.4.2 Código fonte do subsistema de controle do consumo de energia

O código fonte do subsistema de controle inclui as seguintes bibliotecas:

- Wire.h para a comunicação I2C usada pelo LCD;
- LiquidCrystal_I2C.h permite o controle do display LCD via I2C;
- Adafruit_Sensor.h e DHT.h controlam o sensor DHT22 para a leitura de temperatura.

Após a inclusão das bibliotecas necessárias, são definidas as variáveis que controlam o funcionamento do sistema: a variável DHTPIN está associada ao pino 2, onde o sensor DHT22 está conectado; PIRPIN refere-se ao pino 5, conectado ao sensor de movimento PIR; FANPIN está ligada ao pino 3, que envia os sinais de controle para ligar ou desligar o ventilador; e LAMPADAPIN está vinculada ao pino 4, responsável pelo acionamento da lâmpada. A variável DHTTYPE especifica o modelo do sensor de temperatura e umidade, que neste caso é o DHT22, enquanto TEMPERATURA_LIMITE determina o valor de temperatura (em graus Celsius) a partir do qual o ventilador será ativado automaticamente. Ainda nesta seção do código temos a instanciação de objectos: o objecto dht para ler a temperatura do sensor e o objecto lcd para o display de 16 colunas e 2 linhas no endereço I2C 0x27. Temos também a declaração de variáveis de controle para controlar o tempo entre leituras, armazenar o último estado das cargas e evitar leituras ou ações repetidas desnecessárias. Usar essas variáveis evita executar o mesmo comando repetidamente e permite decisões baseadas no tempo.

A segunda parte do código é destinada à função setup onde ocorre:

- Inicialização da comunicação serial;
- Inicialização do sensor DHT22;
- Inicialização do LCD e ligação da sua luz de fundo;
- Configuração dos pinos como INPUT (PIR) e OUTPUT (atuadores). O DHT22 não é configurado como INPUT porque não é necessário, a biblioteca DHT.h já cuida disso internamente.

A terceira parte do código é destinada à função loop. Essa função roda continuamente. Vamos separá-la por blocos:

Leitura da Temperatura (a cada 0,5s):

Nesse bloco lê-se a temperatura e se a temperatura for superior ao limite, liga o ventilador (HIGH), senão, desliga (LOW). Usa-se a variável estadoVentilador para evitar mensagens repetidas no Serial.

```
if (tempoAtual - tempoAnteriorTemp >= intervaloLeitura) {
// Atualiza tempo
temperatura = dht.readTemperature();
// Valida leitura
if (temperatura > TEMPERATURA_LIMITE) {
digitalWrite(FANPIN, HIGH);}
else {
digitalWrite(FANPIN, LOW);
}
}
```

Leitura do Sensor PIR (a cada 5s):

Nesse bloco lê-se o sensor PIR e se detectar movimento (HIGH), liga a lâmpada, senão, desliga (LOW). Usa-se a variável estadoLampada para evitar mensagens repetidas no serial.

```
if (tempoAtual - tempoAnteriorPIR >= intervaloLeitura2) {
int leituraPIR = digitalRead(PIRPIN);
movimentoDetectado = (leituraPIR == HIGH);

if (movimentoDetectado) {
```

```
digitalWrite(LAMPADAPIN, HIGH); }  
else {  
digitalWrite(LAMPADAPIN, LOW);  
}  
}
```

Atualização do Display LCD:

```
lcd.setCursor(0, 0);  
lcd.print("Temp: ");  
lcd.print(temperatura, 1);  
lcd.print((char)223); // símbolo de grau  
lcd.print("C ");  
lcd.setCursor(0, 1);  
lcd.print("Mov: ");  
lcd.print(movimentoDetectado ? "Detectado ": "Ausente ");
```

3.2.5 Página web

A página web permitirá visualizar em tempo real os valores de tensão, corrente, potência activa e consumo de energia. O usuário irá digitar na página a quantidade de energia disponível. Quando o consumo total atingir esse limite, uma notificação será enviada para alertá-lo. Além disso, uma segunda notificação será disparada previamente, quando restarem apenas 25% da energia disponível. Essa notificação avisará sobre o baixo nível de energia e recomendará ao usuário que realize a recarga o quanto antes. O acesso à página será feito por meio de um endereço IP, e para visualizar os dados, o dispositivo deverá estar conectado à mesma rede local do ESP32.

3.2.5.1 Estrutura Geral da Página

A interface é construída utilizando as linguagens HTML, CSS e JavaScript, embutidas diretamente no código-fonte do ESP32 como uma string do tipo `const char[]`. Quando o usuário acessa a rota raiz (`/`), o servidor do ESP32 envia essa página para o navegador.

A estrutura básica do HTML segue o padrão convencional:

```
<!DOCTYPE html>  
<html lang="pt-BR">
```

```
<head>...</head>
<body>...</body>
</html>
```

3.2.5.2 Cabeçalho (<head>)

No cabeçalho, são definidos os metadados da página, o título e os estilos (CSS) para formatação visual:

- `meta charset="UTF-8"`: Permite uso de acentuação e caracteres especiais;
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`: Torna a página responsiva em dispositivos móveis;
- `style`: Define o layout

3.2.5.3 Corpo da Página (<body>)

Título

```
<h1>Sistema de Monitoramento de Energia Eléctrica</h1>
```

Apresenta o nome do sistema no topo da página.

Indicadores em Tempo Real

```
<h2>
```

```
Tensão: <span id="tensao">0</span> V<br>
```

```
Corrente: <span id="corrente">0</span> A<br>
```

```
Potência: <span id="potencia">0</span> kW<br>
```

```
</h2>
```

Esses valores são atualizados dinamicamente via WebSocket, com dados recebidos do ESP32. Cada elemento `` possui um id único que permite o acesso e actualização via JavaScript.

Definição de Limite de Consumo

```
<input type="number" id="limiteEnergia" placeholder="Ex: 10">
```

```
<button id="btnDefinir">Submeter</button>
```

O usuário pode inserir um valor-limite (em kWh) que define a quantidade máxima de energia a ser consumida. Essa informação é armazenada localmente no navegador.

Consumo Acumulado, Limite e Status

```
<div id="consumoAcumulado">0.000</div>
```

```
<div id="limiteAtual">0.000</div>
```

```
<div id="status">Aguardando</div>
```

- consumoAcumulado: Apresenta o total de energia consumida na sessão atual;
- limiteAtual: Mostra o valor definido como limite de consumo;
- status: Informa o estado do sistema (ex: "Conectado", "Desconectado", "Atenção", "Energia cortada").

O progresso do consumo também é representado visualmente por uma barra de progresso horizontal, atualizada conforme o uso de energia.

3.2.5.4 Notificações

```
<div id="notifs" class="log"></div>
```

Apresenta um histórico de mensagens e alertas, como: conexões WebSocket, limites atingidos, redefinições e avisos de consumo.

3.2.5.5 Comunicação com o ESP32 (WebSocket)

A comunicação em tempo real é feita usando WebSockets, que mantêm uma conexão contínua entre o navegador e o ESP32. Essa conexão permite o envio de dados sem a necessidade de atualizações manuais da página.

Inicialização da conexão WebSocket

A conexão é iniciada com:

```
websocket = new WebSocket('ws://' + window.location.hostname + ':81');
```

Conexão e Requisições

- onopen: Exibe status "Conectado" e registra no log;
- onclose: Exibe status "Desconectado", mostra erro visual e tenta reconectar após 3 segundos;
- onmessage: Recebe dados em formato JSON, contendo os valores de tensão, corrente e potência.

3.2.5.6 Lógica de Cálculo do Consumo

A energia consumida (em kWh) é calculada com base na potência instantânea enviada pelo ESP32:

```
const deltaH = (agora - ultimaLeituraMs) / 3600000;
```

```
consumoAcumulado += potencia * deltaH;
```

3.2.5.7 Alertas e Comportamento do Sistema

O sistema emite alertas visuais com base no consumo:

Quando o consumo atinge 75% do limite:

```
if (!aviso75 %% consumoAcumulado >= 0.75 * limiteEnergia) ...
```

Um alerta visual é emitido informando que a energia está quase acabando.

Quando o consumo ultrapassa o limite:

```
if (consumoAcumulado >= limiteEnergia) ...
```

O sistema reseta os dados de consumo, zera os indicadores e alerta o usuário de que o limite de energia foi atingido.

Capítulo 4

Testes e Resultados

Neste capítulo é apresentado o protótipo do sistema e os resultados dos testes realizados nele. Os dois subsistemas definidos nos capítulos anteriores são avaliados separadamente. O primeiro subsistema, responsável pela medição do consumo de energia, é testado com diferentes cargas com o objetivo de verificar se os requisitos estabelecidos para o projeto foram atendidos. No caso do subsistema de controle, seu comportamento é avaliado em pleno funcionamento, analisando-se individualmente o desempenho em cada cômodo do protótipo. Além dos dois subsistemas, também são realizados testes na página web com o objetivo de verificar se os dados são atualizados em tempo real e se o sistema de notificações está funcionando corretamente.

4.1 Protótipo do sistema

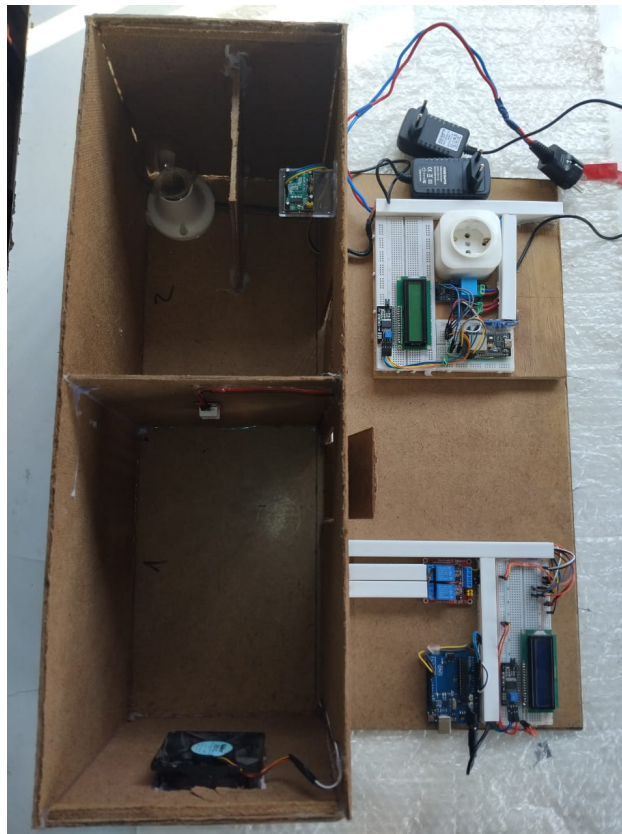


Figura 4.1: Protótipo do sistema . Fonte: Autor

A figura acima ilustra o protótipo do sistema. O processo de montagem envolveu além dos materiais mencionados , o uso de madeira para o suporte do protótipo , calha para os fios, protoboards, cabos *Tereftalato de Polibutileno - Polybutylene Terephthalate (PBT)* e uma tomada que serve de saída alternativa do medidor.

4.2 Testes no subsistema de medição do consumo de energia

Para a demonstração do comportamento do subsistema de medição, foi utilizado um ferro de passar roupa da marca Philips, modelo TYPE HD 1172. O aparelho opera com tensão entre 220 e 240 V e possui potência nominal variando de 950 a 1100 W.

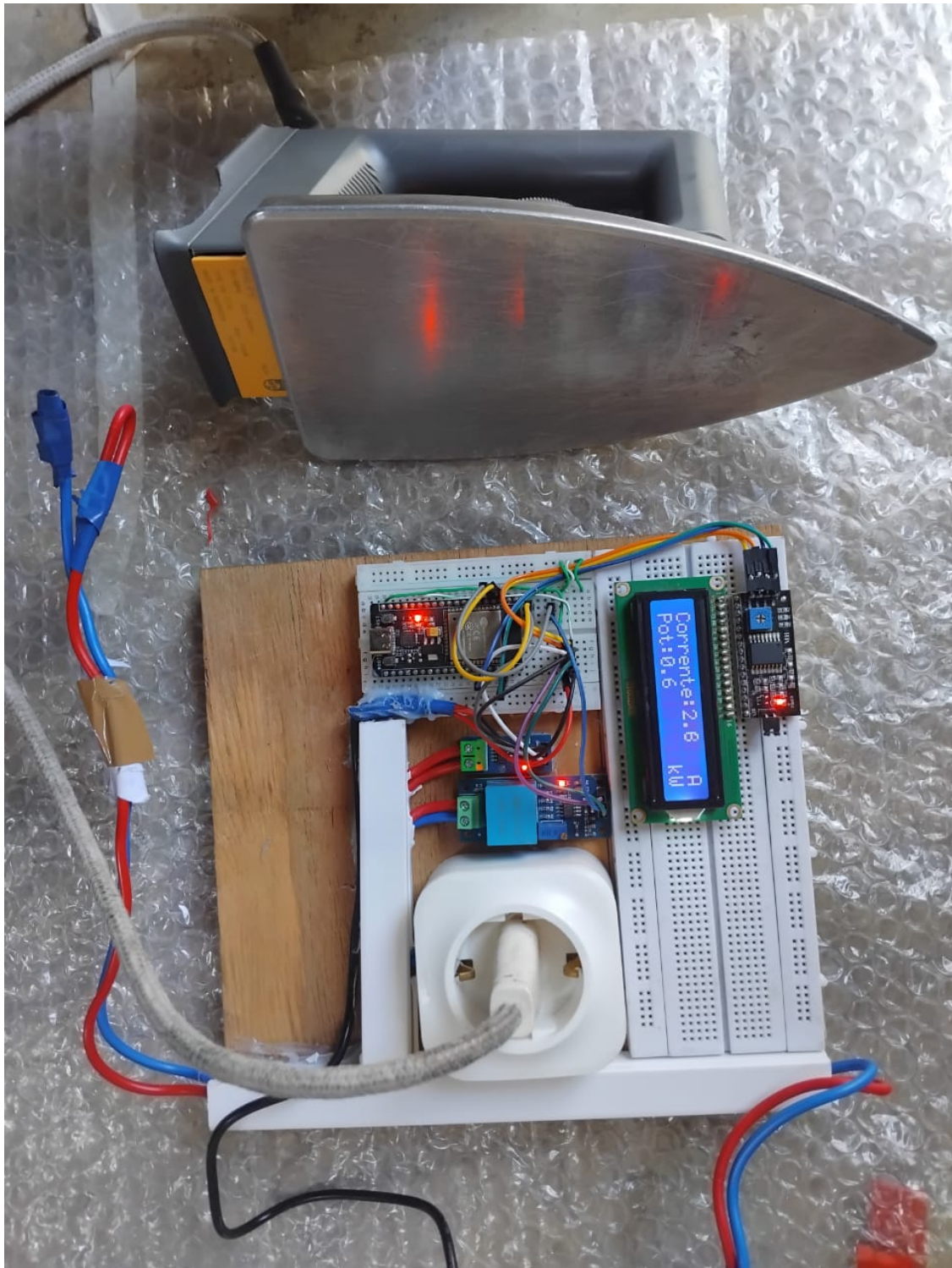


Figura 4.2: Valores de Corrente (A) e Potência (kW) no Display LCD. Fonte: Autor

A figura 4.2 apresenta os valores registrados subsistema de medição, indicando uma potência de 0,6 kW. Esse valor está abaixo da potência nominal do equipamento o que implica que o dispositivo está operando de forma segura. O funcionamento acima da potência nominal pode ocasionar falhas ou danos ao equipamento, uma vez que ele não foi projetado para suportar sobrecargas.

4.3 Testes no subsistema de controle do consumo de energia

4.3.1 Testes no cómodo da carga de ventilação

O ventilador é acionado automaticamente quando a temperatura medida pelo sensor DHT22 ultrapassa 33 °C. Abaixo desse valor, o ventilador permanece desligado. Para simular o aumento de temperatura e testar o funcionamento do sistema, utilizou-se um ferro de passar roupa para aquecer o sensor.

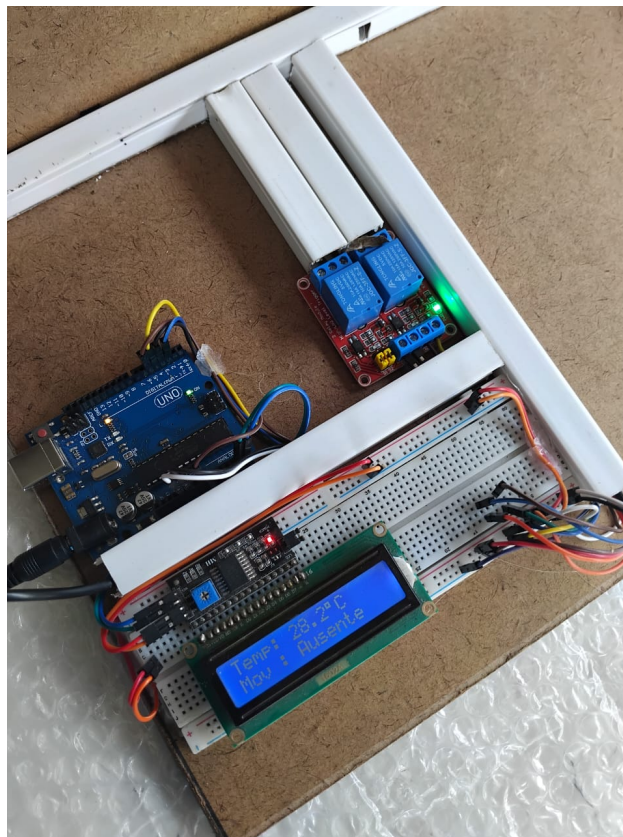


Figura 4.3: Estado do cómodo com a temperatura abaixo do valor de acionamento: (a) Valor da temperatura, (b) Ventilador desligado. Fonte:Autor



Figura 4.4: Sensor DHT22 sendo aquecido com base em um ferro de passar roupa.

Fonte: Autor

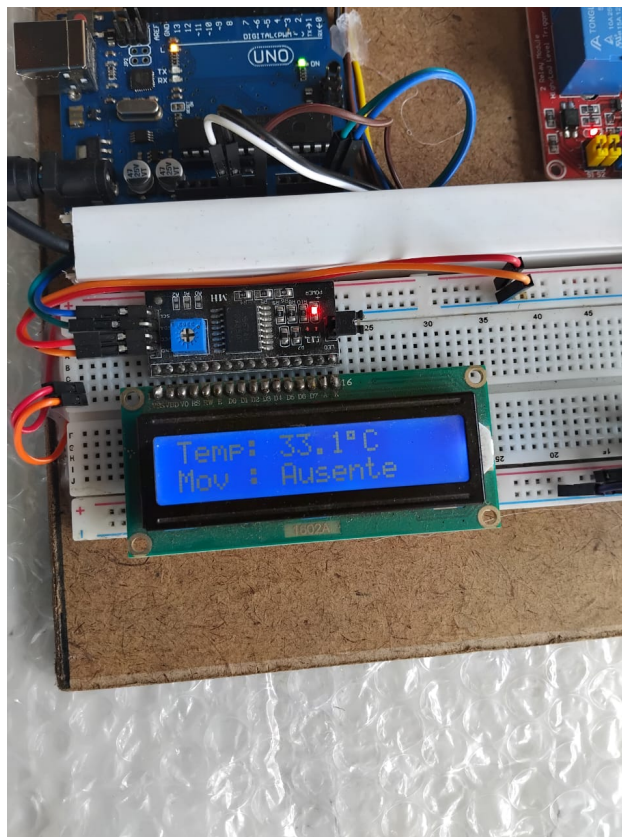


Figura 4.5: A temperatura ultrapassou o limite e por isso o ventilador é acionado: (a) Ventilador acionado, (b) Valor de temperatura acima da temperatura limite. Fonte: Autor

4.3.2 Testes no cómodo da carga de iluminação

A lâmpada é acionada automaticamente quando o sensor PIR detecta movimento no cómodo. Para fins de teste, foi configurado um tempo de activação de 3 segundos.

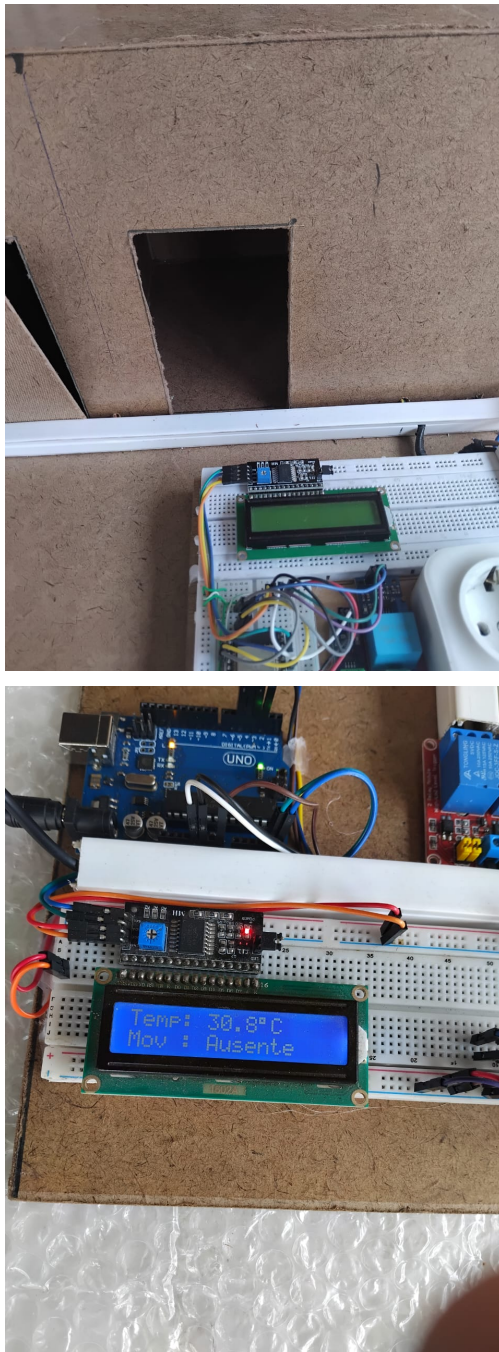


Figura 4.6: Lâmpada apagada pois não foi detectado movimento: (a) Lâmpada apagada, (b) Visualização de movimento ausente no Display LCD. Fonte:Autor

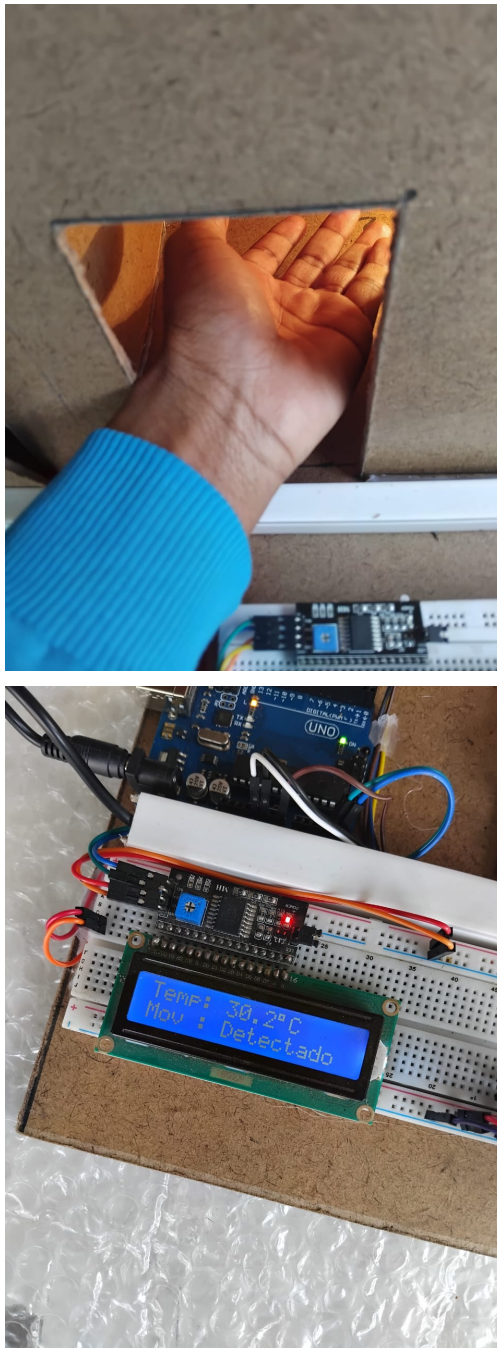


Figura 4.7: Lâmpada acesa pois foi detectado movimento: (a) Lâmpada acesa, (b) Visualização de movimento detectado no Display LCD. Fonte:Autor

4.4 Testes na página web

A figura a seguir apresenta os valores de tensão, corrente, potência e consumo registrados durante um determinado período dos testes iniciais realizados com o ferro de passar roupa, visualizados por meio da página web.



Figura 4.8: Valores de Tensão (V), Corrente (A), Potência (kW), Consumos acumulado e limite (kWh) na página web. Fonte: Autor

4.4.1 Notificações e entrada de dados

O usuário informará na página a quantidade de energia que tem disponível. Quando o consumo acumulado alcançar esse valor, será enviada uma notificação para alertá-lo. Além disso, uma notificação adicional será emitida antecipadamente quando restarem apenas 25% da energia informada, avisando sobre o nível baixo de energia e recomendando que a recarga seja feita o quanto antes. Para fins de teste, foi inserida uma quantidade de energia disponível igual a 0.006kWh.



Figura 4.9: Definindo o valor de 0.006kWh como Energia limite. Fonte: Autor

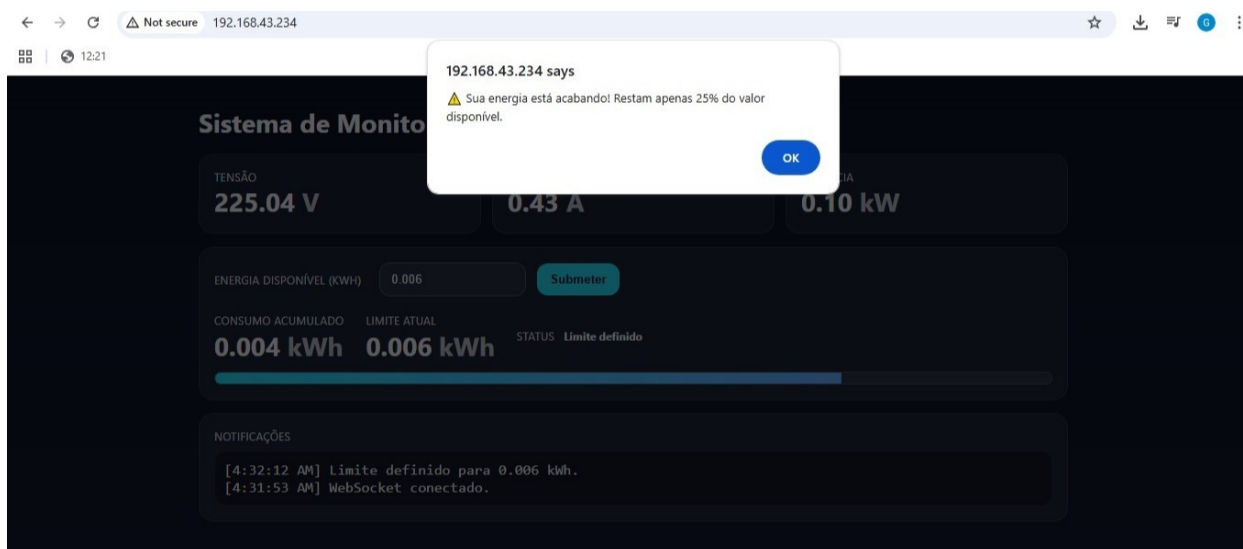


Figura 4.10: Notificação 1: Consumo acumulado atingiu 75% da Energia disponível. Fonte: Autor

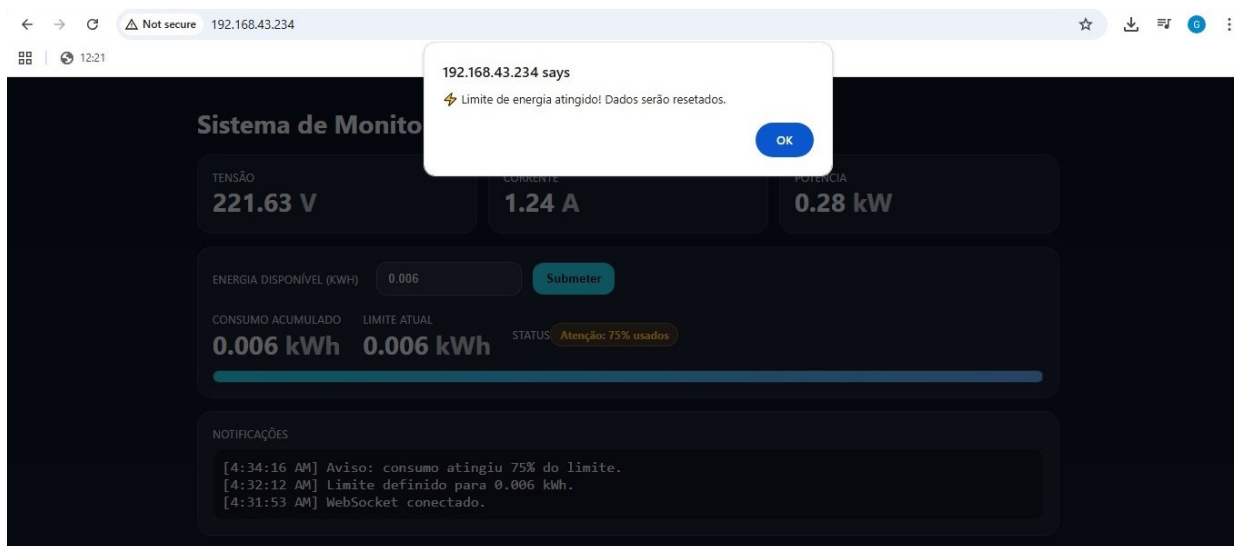


Figura 4.11: Notificação 2: Consumo acumulado atingiu o valor da Energia disponível.

Fonte: Autor

Capítulo 5

Conclusão

A presente monografia propôs um sistema para medição e controle do consumo de energia eléctrica, atendendo a uma demanda crescente por soluções eficientes e sustentáveis. Para alcançar esse propósito foi realizada uma análise detalhada das diversas formas de medição e controle do consumo energético, o que permitiu compreender as principais tecnologias disponíveis e suas aplicações práticas. Entre as soluções analisadas, destacam-se os medidores baseados em shunt resistivo, transformadores de corrente (TC), sensores de efeito Hall, além de sistemas de medição com módulos integrados de energia. No que diz respeito ao controle, foram consideradas abordagens como o uso de temporizadores, sensores ambientais (como temperatura, luminosidade e presença), e atuadores conectados a microcontroladores.

Com base nessa análise, foram identificados os requisitos fundamentais para a implementação de um sistema integrado, que assegura tanto a precisão na medição quanto a efectividade no controle do consumo de energia. Para o subsistema de medição, foram utilizados o sensor de corrente ACS712, o sensor de tensão ZMPT101B e o microcontrolador ESP32, componentes que permitem a aquisição precisa e em tempo real dos dados de consumo eléctrico. Já no subsistema de controle, foram empregados o sensor de temperatura e humidade DHT22, o sensor de presença PIR, o Arduino Uno e um módulo relé de 2 canais, possibilitando a atuação automatizada sobre cargas eléctricas com base em condições ambientais e presença de pessoas. Com esses elementos, projectou-se os dois subsistemas (medição e controle), os quais trabalham de forma conjunta para promover a gestão eficiente do consumo energético.

O desenvolvimento desses subsistemas proporcionou uma solução completa, capaz de monitorar o consumo em tempo real e atuar de forma eficiente para promover economia e

sustentabilidade. Assim, o sistema proposto demonstra potencial para ser aplicado em diversos contextos, contribuindo para a melhoria da eficiência energética e para a redução dos custos associados.

5.1 Recomendação para trabalhos Futuros

Explorar ainda mais a integração do HTML com o ESP32 com o objectivo da página web ser acessada também por dispositivos fora da rede local por meio da Internet.

Referências Bibliográficas

1. “30A Range Current Sensor Module ACS712” (2025). Em: URL: <https://rajguruelectronics.com/Product/165/30A%5C%20range%5C%20Current%5C%20Sensor%5C%20Module%5C%20ACS712.pdf>.
2. “ACS712” (fev. de 2024). Em: URL: <https://www.allegromicro.com/-/media/files/datasheets/acs712-datasheet.ashx>.
3. “Actuators - What is an Actuator?” (2025). Em: URL: <https://www.firgelliauto.com/pt/pages/actuators?srsId=AfmBOopZ-u03nzYfGPt23ebyIVj9llkiOPavUDFDUgbbcvNBVrs0o7Sx>
4. Alves, Mário Ferreira (fev. de 1999). “ABC DOS CIRCUITOS ELÉTRICOS EM CORRENTE ALTERNADA”. Em.
5. Andrade, Elisabeth (2017). *PRINCÍPIOS DE PROGRAMAÇÃO*. Nairobi-Kênia: Universidade Virtual Africana.
6. BARRETO, GUILHERME PIMENTA (2015). “SISTEMA DE MONITORAMENTO E CONTROLE DE CONSUMO DE ENERGIA ELÉTRICA PARA UMA RESIDÊNCIA”. Orientador: Prof. MSc. Luciano Henrique Duque. Trabalho de Conclusão de Curso. Brasília: CENTRO UNIVERSITÁRIO DE BRASÍLIA - UNICEUB.
7. Bayer, Fernando Mariano e Olinto César Bassi de Araújo (2011). *Controle Automático de Processos*. 3ª ed. Santa Maria-RS: e-Tec Brasil.
8. “C++” (dez. de 2020). Em: URL: <https://pt.wikipedia.org/w/index.php?title=C%5C%2B%5C%2B&oldid=70028338>.
9. “Como funciona um Sensor de Movimento PIR – Passive Infrared” (dez. de 2018). Em: URL: <https://www.bosontreinamentos.com.br/eletronica/como-funciona-um-sensor-de-movimento-pir-passive-infrared/>.

10. "DHT22 Datasheet" (2018). Em: URL: <https://www.alldatasheet.com/html-pdf/1132459/ETC2/DHT22/1153/10/DHT22.html>.
11. "Diferença entre tensão e corrente elétrica" (jul. de 2023). Em: URL: <https://alugagera.com.br/noticias/diferenca-tensao-corrente-eletrica>.
12. Feldmann, Lisa et al. (jun. de 2023). "Situação da Eletricidade em Moçambique". Em: URL: https://energypedia.info/wiki/Situa%C3%A7%C3%A3o_da_Eletricidade_em_Mo%C3%A7ambique.
13. Fonseca, Fabrício Ramos da (fev. de 2006). "Sensores". Em: URL: <https://www.adororobotica.com/Sensores.pdf>.
14. "How to Connect ESP32 with WiFi Using Arduino IDE" (2022). Em: URL: <https://linuxhint.com/esp32-wifi-arduino-ide/>.
15. Humberto, José (2025). "O que é web". Em: URL: <https://professorjosehumberto.com.br/glossario/o-que-e-web-entenda-o-conceito-e-sua-importancia/>.
16. "Linguagem de alto nível vs linguagem de baixo nível: definições e diferenças!" (Ago. de 2022). Em: URL: <https://blog.betrybe.com/linguagem-de-programacao/linguagem-alto-e-baixo-nivel/>.
17. Makiyama, Marcio (nov. de 2023). "Placa ESP32: Descubra o que é, para que serve e muito mais!" Em: URL: <https://victorvision.com.br/blog/placa-esp32/>.
18. "Medição de Energia Elétrica" (abr. de 2023). Em: URL: <https://www.ifell.com.br/medicao-de-energia-eletrica/>.
19. "Medidor de eletricidade" (jan. de 2007). Em: URL: https://pt.wikipedia.org/w/index.php?title=Medidor%5C_de%5C_energia%5C_el%5C%C3%A9trica&action=info.
20. "Medidor de energia elétrica – Conheça os tipos e suas principais características" (2025). Em: URL: <https://ims.ind.br/medidor-de-energia-eletrica-conheca-os-tipos-e-suas-principais-caracteristicas/>.
21. "Medidor de tensão e corrente AC com ESP32 e Arduino" (abr. de 2025). Em: URL: <https://www.fernandok.com/2022/04/medidor-de-tensao-e-corrente-ac-com.html>.

22. "MODULO RELE 5V 1 CANAL" (2025). Em: URL: <https://www.mactronica.com.co/modulo-rele-1-canal>.
23. "Motion detection using PIR (Passive infrared sensor)" (dez. de 2018). Em: URL: <https://forumautomation.com/t/motion-detection-using-pir-passive-infrared-sensor/5485>.
24. Nise, Norman S. (2013). *Engenharia de Sistemas de Controle*. Rio de Janeiro: Livros Técnicos e Científicos Editora Ltda. ISBN: 978-85-216-2136-2.
25. "O que é Arduino?" (2025). Em: URL: <https://www.arduinoportugal.pt/o-que-e-arduino/>.
26. "O que é uma linguagem de programação e quais os tipos existem?" (Mai. de 2025). Em: URL: <https://pingback.com/br/resources/linguagem-de-programacao/>.
27. Ogata, Katsuhiko (2010). *ENGENHARIA DE CONTROLE MODERNO*. 5ª ed. São Paulo: Pearson Prentice Hall. ISBN: 978-85-4301-375-6.
28. Oliveira Braz Junior, Osmar de e Marcelo Medeiros (2015). *Fundamentos de Programação Web*. Santa Clara: UnisulVirtual. ISBN: 978-85-7817-841-3.
29. "PIR Motion Sensor" (jan. de 2025). Em: URL: <https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf>.
30. "Potência Elétrica: o que é, fórmula e definição de potência nominal" (set. de 2023). Em: URL: <https://querobolsa.com.br/enem/fisica/potencia-eletrica>.
31. "Projeto Medindo Corrente com o Sensor ACS712 e o Arduino" (2024). Em: URL: <https://www.usinainfo.com.br/blog/projeto-medindo-corrente-com-o-sensor-acs712-e-o-arduino/>.
32. Salite, Daniela, Matthew Cotton e Joshua Kirshner (dez. de 2020). "Acesso à eletricidade e sustentabilidade social em Moçambique". Em.
33. Segundo, Alan Kardek Rêgo e Cristiano Lúcio Cardoso Rodrigues (2015). *Eletricidade em CA*. Minas Gerais: e-Tec Brasil. ISBN: 978-85-68198-03-2.
34. "Sensor" (mai. de 2025). Em: URL: <https://pt.wikipedia.org/wiki/Sensor>.

35. "Sensor de Tensão AC 0 a 250V Voltímetro ZMPT101B" (2024). Em: URL: <https://www.casadarobotica.com/sensores-modulos/sensores/tensao/sensor-de-tensao-ac-0-a-250v-voltimetro-zmpt101b>.
36. "Sensor de Umidade e Temperatura DHT22" (2025). Em: URL: <https://www.makerhero.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/?srsltid=AfmBOoqXUNqAuAS>
37. Sousa, Bruno Jefferson de, José Jorge Lima Dias Junior e André de Araújo Formiga (2014). *Introdução à Programação*. João Pessoa: Editora da UFPB.
38. Souza Guedes, José Carlos de (ago. de 2011). *Manual de Tarifação da Energia Elétrica*. Rio de Janeiro: ELETROBRAS PROCEL.
39. "UNO R3" (2025). Em: URL: <https://docs.arduino.cc/hardware/uno-rev3/#tech-specs..>
40. Vasconcelos, Filipe Matos de (2017). *Geração, transmissão e distribuição de energia elétrica*. Avenida Paris, 675 – Parque Residencial João Piza: Editora e Distribuidora Educacional S.A. ISBN: 978-85-522-0179-3.
41. "What is an Arduino?" (2025). Em: URL: <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>.
42. "ZMPT101B AC Voltage Transformer Sensor Module" (ago. de 2024). Em: URL: <https://datacapturecontrol.com/articles/io-components/sensors/voltage/zmpt101b-ac-voltage-transformer-sensor-module>.

Anexos

Anexo 1

Código 1

1.1 Código do medidor

INCLUSÃO DE BIBLIOTECAS

```
include "EmonLib.h"
```

```
EnergyMonitor emon1;
```

```
include <WiFi.h>
```

```
include <WebServer.h>
```

```
include <WebSocketsServer.h>
```

```
include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

Funções declaradas antecipadamente

```
void handleRoot();
```

```
void websocketEvent(uint8_t num, WStype_t type, uint8_t *payload, size_t wlength);
```

```
void setup() {
```

```
Serial.begin(9600); //INICIALIZACAO DO MONITOROR SERIAL
```

INICIALIZAÇÃO DOS PINOS DO ESP32

```
pinMode(LED, OUTPUT);
```

```
//Fonte: EmonLibrary examples openenergymonitor.org, Licence GNU GPL V3
```

```
emon1.current(36, 2.0);
```

```
emon1.voltage(39, 115, 1.7);
```

INICIALIZAÇÃO NO LCD

```
lcd.init();
```

```

lcd.backlight();
lcd.clear();
CONEXÃO A REDE WIFI//Fonte em URL:https://blog.eletrogate.com/introducao-ao-html-integracao-com-esp32/
WiFi.begin(ssid, password);
while(WiFi.status() != WLZ_CONNECTED){
Serial.print("."); delay(500); //IMPRIME . ENQUANTO TENTAR CONECTAR }
WiFi.mode(WIFI_STA);
Serial.print("Local IP: "); // IMPRIME O IP NO MONITOR SERIAL
Serial.println(WiFi.localIP());
ESCRITA NO LCD
lcd.setCursor(0, 0);
lcd.print("Conectado a Rede!");
lcd.setCursor(2, 1);
lcd.print(WiFi.localIP()); //IMPRIME O IP NO MONITOR SERIAL
delay(10000);
lcd.clear();
//Fonte em URL:https://renatoaloi.blogspot.com/2016/05/codigo-basico-do-web-server-no-esp8266.html
server.on("/", handleRoot);
server.begin();
//Fonte em URL:https://www.youtube.com/watch?v=fREqfdCphRA
WebSocket.begin();
WebSocket.onEvent(WebSocketEvent);}

void loop(){
CHAMADA DE FUNÇÕES
//Fonte em URL:https://www.youtube.com/watch?v=fREqfdCphRA
WebSocket.loop();
Fonte em: https://blog.eletrogate.com/introducao-ao-html-integracao-com-esp32/
server.handleClient();
//Fonte: EmonLibrary examples openenergymonitor.org, Licence GNU GPL V3
CÁLCULO DE CORRENTE
double Irms = emon1.calcIrms(1480);

```

CÁLCULO DE TENSÃO

```
emon1.calcVI(20, 100);
```

```
float supplyVoltage = emon1.Vrms;
```

FILTRAGEM DE VALORES PEQUENOS

```
if (supplyVoltage < 50)
```

```
supplyVoltage = 0.00;}
```

```
if (Irms < 0.55) {
```

```
Irms = 0.00;}
```

CÁLCULO DE POTÊNCIA

```
float pot = (supplyVoltage * Irms) / 1000;
```

ATRIBUICAO DOS VALORES CALULADOS A VARIAVEIS DO TIPO STRING

```
String corr = String(Irms);
```

```
String tens = String(supplyVoltage);
```

```
String potenc = String(pot);
```

ESCRITA NO DISPLAY

```
lcd.setCursor(0, 0);
```

```
lcd.print("Corrente:" + String(Irms, 1) + );
```

```
lcd.setCursor(15, 0);
```

```
lcd.print("A");
```

```
lcd.setCursor(0, 1);
```

```
lcd.print("Pot:" + String(pot, 1) + );
```

```
lcd.setCursor(14, 1);
```

```
lcd.print("kW");
```

Fonte em: <https://blog.eletrogate.com/introducao-ao-html-integracao-com-esp32>

```
void handleRoot(){
```

```
server.send(200, "text/html", webpageCont);}
```

```
void websocketEvent(uint8_t num, WStype_t type, uint8_t *payload, size_t wlength)
```

```
{
```

```
String payloadString = (const char *)payload;
```

```
if (type == WStype_TEXT) //receive text from client
```

```
{
```

```
byte separator = payloadString.indexOf('=');
```

```
String var = payloadString.substring(0, separator);  
String val = payloadString.substring(separator + 1);
```

```
if (var == "LEDonoff") {  
    LEDonoff = false;  
    if (val == "ON") LEDonoff = true;  
}  
}  
}
```

Anexo 2

Código 2

2.1 Código da página web

```
const char webpageCont[] PROGMEM = R"=====(
<!DOCTYPE html>
<html lang="pt-BR">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
<title>Gráficos de Potência e Consumo</title>
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<style>
dynRectangle1 {
width:300px;
height:12px;
top: 9px;
background-color: rgba(255, 0, 0, 0.863);
z-index: -1;
margin-top: 1px;
border: 3px solid black;}
dynRectangle2 {
width:200px;
height:12px;
```

```

top: 9px;
background-color:rgba(0, 0, 255, 0.534);
z-index: -1;
margin-top:1px;
border: 3px solid black;}

body {
font-family: Arial, sans-serif;
text-align: center;
margin: 20px;}

table{
margin: 20px auto;
border: 1px solid black;
border-collapse: collapse;
width: 80%;}

th, td {
border: 1px solid black;
padding: 8px;
text-align: center;}

graficoConsumo {
width: 80%;
margin: 20px auto;}

buttons {
margin: 20px 0;}

input[type="number"] {
padding: 5px;
margin: 10px;
width: 60px;
text-align: center;}

</style>
</head>
<body>
<h1>Consumo de Energia Elétrica Numa Casa</h1>
<h2>
Tensão: <span style="color:rgb(216, 3, 3)"id="tensao»0</span>

```

```

V<br>
</h2>
<h2>
Corrente: <span style="color:rgb(216, 3, 3)"
id="corrente»0</span> A<br>
</h2>
<h2>
Potência: <span style="color:rgb(216, 3, 3)"
id="potencia»0</span> KW<br>
</h2>
<hr>
<table>
<tr>
<th>Tempo</th>
<th>Tensão (U)</th>
<th>Corrente (i)</th>
<th>Potência (W)</th>
<th>Consumo (Wh)</th>
</tr>
<tbody id="data-table»
</tbody>
</table>

<br>
<hr>
<h2>Gráfico de Consumo Diário</h2>
<input type="number" id="numDias" min="1" max="30" placeholder="Número de dias (1 a
30)"/>
<div id="buttons»
<button id="gerarGrafico»Gerar Gráfico</button>
<button id="limparGrafico»Limpar Gráfico</button>
</div>
<canvas id="graficoConsumo»</canvas>

```

```

<script>
// Valores de consumo diário já inicializados
const consumoDiario = [
3.5, 3.7, 3.8, 3.4, 3.9, 3.5, 4.0,
4.3, 4.5, 3.8, 4.3, 3.9, 4.1, 3.8,
3.4, 3.9, 4.5, 3.3, 4.2, 3.6, 3.7,
4.4, 4.0, 4.2, 3.6, 3.9, 4.1, 3.8,
4.0, 4.1, 4.3
];

let graficoConsumo;
let labelsConsumo = Array.from( length: 30 , (_, i) => 'Dia {i + 1}');
let consumoAtual = [...consumoDiario];
function gerarGraficoConsumo(numDias) {
const ctxConsumo = document.getElementById('graficoConsumo').getContext('2d'); ctx-
Consumo.clearRect(0, 0, ctxConsumo.canvas.width, ctxConsumo.canvas.height); // Limpa
o canvas antes de desenhar
graficoConsumo = new Chart(ctxConsumo, {
type: 'line', // Gráfico linear
data: {
labels: labelsConsumo.slice(0, numDias), // Eixo X: Dias datasets: [{
label: 'Consumo Diário (kWh)',
data: consumoAtual.slice(0, numDias), // Pega apenas os dados dos dias selecionados
backgroundColor: 'rgba(75, 192, 192, 0.2)',
borderColor: 'rgba(75, 192, 192, 1)',
borderWidth: 2,
fill: true // Preenche a área sob a linha}}],
options: {
scales: {
y: {
beginAtZero: true,
title: {
display: true,
text: 'Consumo (kWh)'

```

```
}}}}}); }
```

```
// Limpar gráfico e permitir novo número de dias  
document.getElementById('limparGrafico').addEventListener('click', () => {  
  const ctxConsumo = document.getElementById('graficoConsumo').getContext('2d'); ctx-  
Consumo.clearRect(0, 0, ctxConsumo.canvas.width, ctxConsumo.canvas.height); // Limpa  
o canvas
```

```
document.getElementById('numDias').value = ""; // Limpa o campo de entrada  
graficoConsumo graficoConsumo.destroy(); // Destroi o gráfico existente, se houver });
```

```
// Gerar gráfico quando o número de dias for fornecido  
document.getElementById('gerarGrafico').addEventListener('click', () => { const numDias  
= parseInt(document.getElementById('numDias').value);
```

```
if (numDias < 1 || numDias > 30 || isNaN(numDias)) {  
  alert("Por favor, insira um número válido de dias entre 1 e 30.");  
  return; }
```

```
gerarGraficoConsumo(numDias); // Chama a função para gerar o gráfico });
```

```
InitWebSocket();
```

```
function InitWebSocket() {  
  websocket = new WebSocket('ws://' + window.location.hostname + ':81/');  
  websocket.onmessage = function(evt) {  
    let JSONobj = JSON.parse(evt.data);  
    console.log(JSONobj);
```

```
let now = new Date();  
let timeString = now.toLocaleTimeString();
```

```
let voltage = parseFloat(JSONobj.HUM).toFixed(2);  
let current = parseFloat(JSONobj.TEMP).toFixed(2);  
let power = parseFloat(JSONobj.POT).toFixed(2);
```

```

// Cálculo de consumo acumulado
consumption += parseFloat(power) / 1800;

// Atualização dos dados na tabela
updateTable(timeString, voltage, current, power, consumption.toFixed(4));

// Atualiza o consumo diário updateDailyConsumption(power); }

function updateTable(time, voltage, current, power, consumption) { const tableBody =
document.getElementById('data-table');
let row = document.createElement('tr');

row.innerHTML = '
<td>time</td>
<td>voltage</td>
<td>current</td>
<td>power</td>
<td>consumption</td>
'; tableBody.appendChild(row); }

function updateDailyConsumption(power) {
let dailyConsumption = consumoAtual[consumoAtual.length - 1] || 0; consumoAtual.push(dailyConsumption
+ parseFloat(power));

if (consumoAtual.length > 30) { consumoAtual.shift();
} }
</script>
</body>
</html>

)=====";

```

Anexo 3

Código 3

3.1 Código do sistema de controle

```
// Inclusao de Bibliotecas
include <Wire.h>
include <LiquidCrystal_I2C.h>
include <Adafruit_Sensor.h>
include <DHT.h>

// Configurações de pinos e sensor
define DHTPIN 2
define DHTTYPE DHT22
define PIRPIN 5
define FANPIN 3
define LAMPADAPIN 4
define TEMPERATURA_LIMITE 35.0 // Ajuste conforme necessário

DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27, 16, 2); // Verifique o endereço I2C (pode ser 0x3F)

unsigned long tempoAnteriorTemp = 0;
unsigned long tempoAnteriorPIR = 0;
const unsigned long intervaloLeitura = 500; // 0.5 segundos
const unsigned long intervaloLeitura2 = 5000; // 2 segundos
```

```

float temperatura = 0;
bool movimentoDetectado = false;
bool estadoLampada = false;
bool estadoVentilador = false;

void setup() {
Serial.begin(9600);
dht.begin();
lcd.init();
lcd.backlight();

pinMode(PIRPIN, INPUT);
pinMode(FANPIN, OUTPUT);
pinMode(LAMPADAPIN, OUTPUT);

lcd.setCursor(0, 0);
lcd.print("Iniciando...");
delay(1000);
lcd.clear();
}

void loop() {
unsigned long tempoAtual = millis();

// Leitura de temperatura
if (tempoAtual - tempoAnteriorTemp >= intervaloLeitura) {
tempoAnteriorTemp = tempoAtual;

temperatura = dht.readTemperature();
if (isnan(temperatura)) {
Serial.println("Erro ao ler o sensor DHT22!"); }
else {
Serial.print("Temperatura atual: ");
Serial.print(temperatura);

```

```

Serial.println("°C");

    if (temperatura > TEMPERATURA_LIMITE) {
digitalWrite(FANPIN, HIGH);
if (!estadoVentilador) {
Serial.println(«» Temperatura alta! Ventilador LIGADO.");
estadoVentilador = true;
}
}
else {
digitalWrite(FANPIN, LOW);
if (estadoVentilador) {
Serial.println(«» Temperatura normal. Ventilador DESLIGADO.");
estadoVentilador = false;
}
}
}
}

// Leitura do PIR
if (tempoAtual - tempoAnteriorPIR >= intervaloLeitura2) {
tempoAnteriorPIR = tempoAtual;

    int leituraPIR = digitalRead(PIRPIN);
movimentoDetectado = (leituraPIR == HIGH);

    if (movimentoDetectado) {
digitalWrite(LAMPADAPIN, HIGH);
if (!estadoLampada) {
Serial.println(«» Movimento detectado! Lâmpada ACESA.");
estadoLampada = true;
}
}
else {

```

```
digitalWrite(LAMPADAPIN, LOW);
if (estadoLampada) {
  Serial.println(»» Sem movimento. Lâmpada APAGADA.");
  estadoLampada = false;
}
}
}

    // Atualizar display LCD
lcd.setCursor(0, 0);
lcd.print("Temp: ");
lcd.print(temperatura, 1);
lcd.print((char)223); // símbolo de grau
lcd.print("C ");

    lcd.setCursor(0, 1);
lcd.print("Mov: ");
lcd.print(movimentoDetectado ? "Detectado ": "Ausente ");
}
```