



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
CENTRO MULTIDISCIPLINAR CARAÚBAS
DEPARTAMENTO DE ENGENHARIAS
CURSO DE ENGENHARIA ELÉTRICA

JOÃO UDENILSON DE OLIVEIRA JÚNIOR

**DESENVOLVIMENTO DE UMA INTERFACE PARA MEDIÇÃO, ANÁLISE E
SUPERVISÃO DE CONSUMO DE ENERGIA ELÉTRICA**

Caraúbas/RN

2019

JOÃO UDENILSON DE OLIVEIRA JÚNIOR

DESENVOLVIMENTO DE UMA INTERFACE PARA MEDIÇÃO, ANÁLISE E
SUPERVISÃO DO CONSUMO DE ENERGIA ELÉTRICA

Monografia apresentada a Universidade
Federal Rural do Semi-Árido – UFERSA,
destinado à obtenção do Bacharelado em
Engenharia Elétrica.

Orientador (a): Prof. Dr. Bruno Emmanuel De
Oliveira Barros Luna.

Caraúbas

2019

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº

9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

OO48d Oliveira Júnior, João Udenilson de .
Desenvolvimento de uma interface para
medição, análise e supervisão do consumo de
energia elétrica / João Udenilson de
Oliveira Júnior. 2019.
46 f. : il.

Orientador: Bruno Emmanuel De
Oliveira Barros Luna.
Monografia (graduação) -
Universidade Federal
Rural do Semi-árido, Curso de
Engenharia Elétrica, 2019.

1. Sistema supervisorio. 2. Smart
grid. 3. MQTT. 4. ESP32. I. Luna, Bruno
Emmanuel De Oliveira Barros , orient. II.
Título.

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

JOÃO UDENILSON DE OLIVIRA JÚNIOR

**DESENVOLVIMENTO DE UMA INTERFACE PARA MEDIÇÃO, ANÁLISE E
SUPERVISÃO DO CONSUMO DE ENERGIA ELÉTRICA**

Monografia apresentada a Universidade Federal
Rural do Semi-Árido – UFERSA, destinado à
obtenção do Bacharelado em Engenharia Elétrica.

Orientador (a): Prof. Dr. Bruno Emmanuel De
Oliveira Barros Luna.

APROVADA EM: 03 / 08 / 2019

BANCA EXAMINADORA



Bruno Emmanuel De Oliveira Barros Luna, Prof. Dr. (UFERSA)
Presidente



Herick Talles Queiroz Lemos (Me. em Eng. Elétrica)
Primeiro Membro



Daniel Pablo Dantas Diógenes (Me. em Eng. Elétrica)
Segundo Membro

RESUMO

O objetivo deste trabalho é apresentar o desenvolvimento de um sistema supervisório para monitoramento de energia elétrica. Para isso, utiliza-se um microcontrolador ESP32, o qual por meio de um programa, desenvolvido em linguagem C, possibilita comunicação com sensores de corrente, tensão com a rede local através do protocolo MQTT. Desenvolve-se uma interface *web* de fácil interação que proporcionará ao consumidor informações adicionais ao consumo diário. Finalmente, construção do protótipo do sistema monitoramento, onde foram realizados testes com diferentes tipos de carga a fim de constatar a confiabilidade do projeto desenvolvido, comparando os valores obtidos com os dados de aparelhos padrão de medição como multímetro e alicate amperímetro, tais testes permitiram comprovar a funcionalidade do sistema com uma aproximação aceitável.

Palavras chaves: Sistema supervisório, *Smart grid*, MQTT, ESP32.

ABSTRACT

The objective of this paper is to present the development of a supervisory system for monitoring electric power. To this end, use an ESP32 microcontroller, which can be developed using the C language, enabling communication with current sensors with a local network using the MQTT protocol. Develop an easy-to-interact web interface that exposes consumers to their daily consumption. Finally, the construction of the prototype monitoring system, where they were tested with different load types of a summer project, comparing the values with standard measurement movement data such as multimeter and ammeter, such as tests tested to prove system functionality. with an acceptable approximation.

Keywords: Supervisory system, Smart grid, MQTT, ESP32.

LISTA DE FIGURAS

Figura 1-Triângulo de potências.....	15
Figura 2-Medidor eletromecânico de indução monofásico.	17
Figura 3-Medidor comercial LUMEN 3.....	18
Figura 4-Conexão <i>smart grid</i>	19
Figura 5- Portas ESP32 WROOM32 DevKit.	21
Figura 6- Sensor SCT-013-000.	22
Figura 7- Interface da ferramenta Node-Red.Fonte: autoria própria.....	23
Figura 8- Arquitetura de rede MQTT.....	24
Figura 9-Eschema de ligação para o TC.	26
Figura 10- Código de calibração utilizado no projeto.	27
Figura 11- Circuito de tensão simulado no PSIM.	28
Figura 12-Circuito de corrente simulado no PSIM.	29
Figura 13- Placa desenvolvida no software Proteus.....	30
Figura 14- Placa corroída e com seu componentes fixados.	30
Figura 15- Nó de debug.....	31
Figura 16- Nó MQTT.....	31
Figura 17- Nó file.....	31
Figura 18- Nó gauge.....	32
Figura 19- Nó chart.	32
Figura 20- Página <i>web</i> de edição.....	32
Figura 21- Interface <i>web</i> do Node-Red.	33
Figura 22- Medição da tensão e corrente da lâmpada de 150W com o alicate amperímetro.	34
Figura 23-Amostra da interface do sistema supervisorío com valores equivalente do experimento 1.....	35
Figura 24- Medição da tensão e corrente do ventilador com o alicate amperímetro.....	36
Figura 25-Amostra da interface do sistema supervisorío com valores equivalente do experimento.....	37
Figura 26- Medição da tensão e corrente da lâmpada incandescente e ventilador com o alicate amperímetro.....	38
Figura 27- Amostra da interface do sistema supervisorío com valores equivalente do experimento 3.....	38

LISTA DE GRÁFICOS

Gráfico 1- Variação da tensão ao aplicar diferentes cargas.Fonte: Autoria própria.....	39
Gráfico 2- Variação da corrente ao aplicar diferentes cargas.....	39
Gráfico 3- Variação da fator de potência ao aplicar diferentes cargas.....	40

Sumário

1	INTRODUÇÃO	11
2	OBJETIVOS	11
3	REFERENCIAL TEÓRICO	12
3.1	Corrente e tensão elétrica	12
3.2	Energia e potência.....	12
3.3	Tipos de potência	13
3.3.1	Potência instantânea.....	13
3.3.2	Potência ativa e potência reativa.....	13
3.3.3	Potência aparente e fator de potência	15
3.4	Sistema de medição de consumo elétrico	16
3.4.1	Medidores eletromecânicos	16
3.4.2	Medidores eletrônicos.....	17
3.5	Smart grid.....	18
3.6	Placa eletrônica esp 32.....	19
3.6.1	Hardware.....	20
3.7	Sensores	21
3.7.1	Tensão.....	21
3.7.2	Corrente	22
3.8	Plataforma node-red.....	23
3.9	Protocolo mqtt.....	23
4	MATERIAIS E METODOS	25
4.1	Primeiros testes	25
4.1.1	Corrente	25
4.1.2	Tensão.....	26
4.2	Programação do microcontrolador.....	26
4.2.1	Código da biblioteca	27
4.3	Placa	28
4.4	Manipulação de dados.....	30
5	RESULTADO E DISCUSSÕES	34
5.1	Teste com lâmpada incandescente 150w	34
5.2	Teste com ventilador 80w.....	36

5.3	Teste com lâmpada incandescente 150w e ventilador de 80w	37
5.4	Teste de variação de cargas	39
6	CONCLUSÃO	41
7	REFERENCIAS	42
8	APÊNDICE A – CÓDIGO FINAL DO SISTEMA SUPERVISÓRIO	44

1 INTRODUÇÃO

O consumo de energia elétrica em escala mundial está aumentando, devido ao constante crescimento populacional, isto implica em maiores investimentos e em uma elevação dos impactos ambientais. No Brasil, os meios pelos quais a energia é gerada hoje apresenta uma grande dependência dos fenômenos ambientais, onde a maior parte da matriz energética é dada pelas hidrelétricas onde a ocorrência de períodos de seca ocasionam uma baixa produção e isso provoca o aumento de tarifas afetando diretamente aos consumidores.

O aumento da eficiência energética se torna essencial para que tais impactos sejam minimizados, sendo ainda uma das maneiras mais rápidas, limpas e eficazes de combater o consumo energético (JEGUNDO, 2012).

Tendo em vista essa modernização surgiram as *Smarts Grids* capazes de supervisionar e controlar os elementos do sistema elétrico de potência através de uma rede de telecomunicação. Se destacam por ter um conceito baseado em monitoramento inteligente, aumentando os ganhos em confiabilidade, eficiência energética, participação dos consumidores, geração de uma energia mais limpa, além de possuir o fluxo de potência de forma bidirecional. Com a transmissão bidirecional, o consumidor pode usar, gerar e vender energia, e ter acesso a informações que possibilita um controle maior sobre seu consumo.

2 OBJETIVOS

Desenvolver um sistema supervisor de energia elétrica, cujo a finalidade é quantificar e determinar onde está sendo consumida a energia elétrica, e através da interface de comunicação, proporcionar o acesso à leitura capaz de analisar as potências aparente, ativa e reativa, bem como o fator de potência, para unidades consumidoras monofásicas onde a corrente elétrica não ultrapasse 25 ampères. E por meio da interpretação dessas informações recebidas, desenvolver previsões e propostas, a fim de proporcionar o melhor uso da energia através desse monitoramento.

3 REFERENCIAL TEÓRICO

Para melhor compreensão, será apresentado alguns conceitos fundamentais para o desenvolvimento deste trabalho.

3.1 Corrente e tensão elétrica

Corrente e tensão elétrica são as grandezas básicas no estudo de sistemas de elétricos. A corrente elétrica é definida como sendo o fluxo de cargas elétricas durante uma unidade de tempo, medida em ampères, já a tensão (ou diferença de potencial) sendo a energia essencial para transportar uma carga através de um caminho, medida em volts. (ALEXANDER E SADIKU, 2013). A definição matemática pode ser representada pelas seguintes equações:

$$I = \frac{\Delta Q}{\Delta t} \quad (1)$$

Onde: I = corrente elétrica em ampères [A];

ΔQ = variação de carga em coulombs [C];

Δt = variação do tempo em segundos [s].

$$V = \frac{W}{Q} \quad (2)$$

Onde: V = tensão em volts [V];

W = energia em joules [J];

Q = cargas em coulombs [C].

3.2 Energia e potência

Segundo Alexander e Sadiku (2013), a energia de um sistema é entendida como a capacidade de realizar trabalho, é medida em joules ou ainda em watt-segundo, já a potência representa a velocidade com que essa energia é consumida ou absorvida, medida em watts. Matematicamente têm-se as equações:

$$W = P \cdot \Delta t \quad (3)$$

Onde: W = energia em joules [J], ou watt-segundo [Ws];

P = potência em watts [W];

Δt = variação do tempo em segundos [s].

$$P = \frac{dW}{dt} = \frac{dW}{dq} \cdot \frac{dq}{dt} = V \cdot I \quad (4)$$

Onde: P = potência em watts [W];

V = tensão em volts [V];

I = corrente elétrica em ampères [A].

A potência é o parâmetro mais importante dentro dos sistemas elétricos, pois, é responsável por transmitir energia entre todos os pontos do circuito, desta maneira todo equipamento elétrico tem indicado qual a potência necessária para que este funcione adequadamente. No caso de ultrapassar a potência nominal pode causar danos ao equipamento. (ALEXANDER E SADIKU, 2013).

3.3 Tipos de potência

Quando trabalhamos em regime de corrente alternada, a potência divide-se em partes com propriedades distintas, portanto é necessário o entendimento de algumas destas.

3.3.1 Potência instantânea

A potência instantânea de um dispositivo ou de um sistema se dá entre o produto da corrente instantânea que flui através deste, e a tensão instantânea que o dispositivo está submetido. Sendo dada em watts. Podendo ser positiva, quando o dispositivo consome potência do sistema, ou negativa, quando o dispositivo fornece potência ao sistema. (ALEXANDER E SADIKU, 2013).

A representação dos valores instantâneos é dada por letras minúsculas, assim temos que a potência instantânea:

$$p(t) = v(t) \cdot i(t) \quad (5)$$

3.3.2 Potência ativa e potência reativa

Ao aplicarmos as equações para tensão e corrente dependentes do tempo, Equação (6) e Equação (7) respectivamente, na equação da potência instantânea (Equação (5)), e aplicarmos a identidade trigonométrica para produto de cossenos, obtemos duas parcelas para potência uma constante e outra que varia no tempo, conforme mostra a Equação (8).

$$v(t) = V_m \cdot \cos(\omega t + \theta_v) \quad (6)$$

$$i(t) = I_m \cdot \cos(\omega t + \theta_i) \quad (7)$$

$$p(t) = \frac{V_m \cdot I_m}{2} \cdot \cos(\theta_v - \theta_i) + \frac{V_m \cdot I_m}{2} \cdot \cos(2\omega t + \theta_v + \theta_i) \quad (8)$$

Onde: V_m e I_m = Tensão máxima [V] e corrente máxima [A] respectivamente;

θ_v e θ_i = Ângulos de fase [°] de tensão e correntes respectivamente;

ω = velocidade angular das ondas de tensão e corrente.

Segundo Alexander e Sadiku (2013) a potência média, dada em watts, é a média de todas as potências instantâneas ao longo de um período de tempo, e é dada pela Equação.

$$P = \frac{1}{T} \int_0^T p(t) \cdot dt \quad (9)$$

Onde: T = Período [s] de utilização da potência;

Substituindo a Equação (8) na Equação (9), obtemos que a potência média é igual a parcela constante da potência instantânea (Equação (10)), a potência ativa é responsável pela realização ativa de trabalho, ou seja, é responsável pela conversão de potência elétrica em outra forma de potência, gerando calor ou movimento.

$$P = \frac{V_m \cdot I_m}{2} \cdot \cos(\theta_v - \theta_i) \quad (10)$$

A potência ativa ainda pode ser tratada em termos de valores eficazes, ou valores RMS, de tensão e corrente, Equações (11) e (12) respectivamente, resultando na Equação (13).

$$V_{RMS} = \sqrt{\frac{1}{T} \int_0^T v^2 \cdot dt} = \frac{V_m}{\sqrt{2}} \quad (11)$$

$$I_{RMS} = \sqrt{\frac{1}{T} \int_0^T i^2 \cdot dt} = \frac{Im}{\sqrt{2}} \quad (12)$$

$$P = V_{RMS} I_{RMS} \cdot \cos(\theta_v - \theta_i) \quad (13)$$

Potência reativa (Q), é a parcela da potência que não é convertida em trabalho útil, sendo armazenada e devolvida ao gerador sendo sua unidade o volt-ampère-reactivo (VAR). Seu valor é dado por:

$$Q = V_{RMS} I_{RMS} \cdot \sin(\theta_v - \theta_i) \quad (14)$$

3.3.3 Potência aparente e fator de potência

Segundo Alexander e Sadiku (2013) a potência ativa é composta pelo produto entre corrente e tensão eficazes e o cosseno da diferença entre seus respectivos ângulos de fase. Sendo o produto da corrente e tensão eficazes o termo conhecido como potência aparente (S), sua unidade é volt-ampère (VA), já o cosseno da diferença entre os ângulos de fase é chamado de fator de potência (FP).

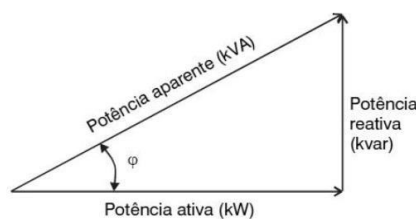
$$S = V_{RMS} \cdot I_{RMS} \quad (15)$$

$$FP = \cos(\theta_v - \theta_i) \quad (16)$$

Através de relações trigonométricas também é possível obter uma equação que relaciona as potências aparente, ativa e reativa. Tal relação é comumente representada pelo triângulo de potências, mostrado na Figura 1, onde ϕ representa a diferença de fases entre tensão e corrente.

$$S = \sqrt{P^2 + Q^2} \quad (17)$$

Figura 1-Triângulo de potências.



Fonte: Martinez, 2013.

3.4 Sistema de medição de consumo elétrico

Para realizar a medição do consumo elétrico de um centro comercial, indústria ou residência, são utilizados instrumentos de medição estes são subdivididos em dois grupos eletromecânicos e eletrônicos, estes dispositivos apresentam o consumo de energia elétrica em um valor quilowatt hora (kWh). Sendo esse valor a potência consumida em um determinado tempo.

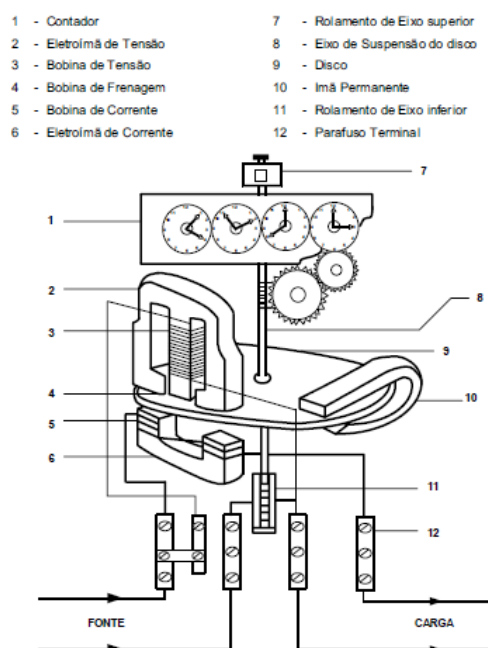
Segundo Kup (2015), o processo de medição de energia elétrica está suscetível a erros de medição por parte do técnico durante a visita a residência do consumidor e ainda pelo responsável do faturamento da conta de energia, visto que o valor consumido medido pelo técnico é anotado à mão.

O sistema utilizado não permite ao consumidor interagir com os serviços prestados pela concessionária fazendo com que não percebam os reflexos decorrentes da forma de usar a eletricidade e tome ações de acordo com as informações recebidas.

3.4.1 Medidores eletromecânicos

Os medidores eletromecânicos possuem design e precisão melhorados, porém possuem o mesmo princípio de funcionamento do pensado por Duncan. Esses medidores trabalham de forma idêntica a um motor de indução, onde os campos provenientes das bobinas de tensão e de corrente provocam a rotação de um disco móvel, que por sua vez aciona um conjunto de engrenagens responsáveis em registrar o consumo da carga. (MORO,2013).

Figura 2-Medidor eletromecânico de indução monofásico.



Fonte: Velasco (2017)

No Brasil, a norma NBR 8277 regulamenta os medidores, essa norma é responsável por especificar o tipo de medidor, materiais, erro admissível, valores de tensão e corrente. De acordo com a norma os medidores devem se encaixar em uma classe de erro, conforme a Tabela 1.

Tabela 1-Limite de erro de acordo com a classe

Classe de exatidão	Limite de erro
2	± 2
1	± 1
0,5	$\pm 0,5$

Fonte: NBR 8377 (1995).

O medidos eletromecânicos utilizados no Brasil se encaixam na classe de exatidão 2, onde admitem erro máximo de 2%. Medidores com classe de exatidão 1 ou 0,5 também podem ser utilizados, porem possuem um elevado valor de compra, pois os mesmos não são fabricados no Brasil.

3.4.2 Medidores eletrônicos

Os medidores eletrônicos possuem mais funcionalidade comparados aos eletromecânicos. O consumo elétrico é mostrado por um display, além de medir valores de potência instantânea, ativa e reativa em um determinado período do dia, como horários de pico. Além da exibição dos valores, também é possível transmitir informações para outros aparelhos eletrônicos de forma remota.

Figura 3-Medidor comercial LUMEN 3.



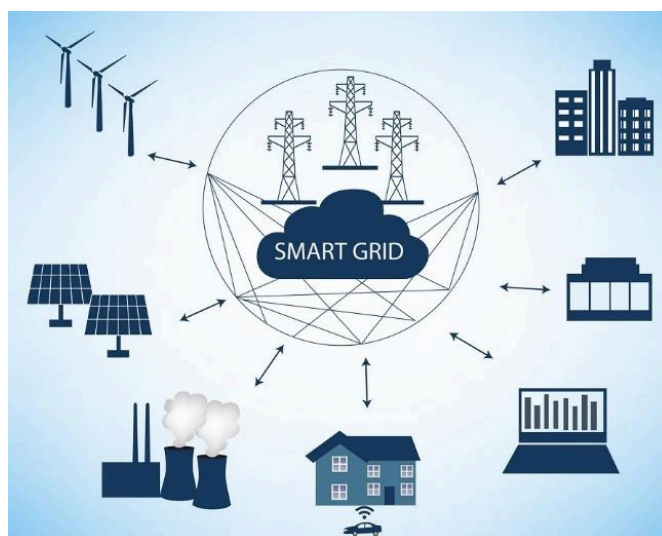
Fonte: NANSEN,2019.

O medidor eletrônico LUMEN 3 da marca Nansen, muito utilizado para consumidores residenciais, pois foca apenas na energia ativa.

3.5 Smart grid

Atualmente o fluxo de energia ocorre de forma radial, onde a energia sai das geradoras e vai para o consumidor. Com a implantação da Smart Grid esse formato muda, pois, o fluxo de energia irá ocorrer de forma mais dinâmica, sendo bidirecional onde o próprio consumidor será capaz de contribuir com o sistema, gerando parte da energia injetada na rede elétrica.

Desse modo todo o sistema consegue trabalhar melhor, conseguindo controlar o fluxo de energia e passando a ser também mais sustentável.

Figura 4-Conexão *smart grid*.

Fonte: MINHA CASA SOLAR, 2019.

As *Smats Grids* surgiram da união dos sistemas de transmissão e distribuição de energia com os recursos comuns ao campo de Tecnologia da Informação. Com o ganho de recursos esses sistemas ganham eficiência, auxiliam na redução do desperdício de energia e possibilitam acompanhar o quanto de eletricidade é gasta. Esses sistemas trabalham com fluxos de comunicação e energia de forma bidirecional, permitindo a entrada de microgeradores de energia (residências com painéis fotovoltaicos), o consumidor pode usar, gerar e vender energia sendo consumidor e produtor de energia ao mesmo tempo além de identificar, quase instantaneamente, problemas em seu sistema (UFJF NOTÍCIAS, 2017).

3.6 Placa eletrônica esp 32

Segundo Vagapov (2017), o ESP32 é uma plataforma de hardware livre ideal para a criação de dispositivos que permitam interação com o ambiente utilizando sensores como entrada, possui módulo WiFi integrado, Bluetooth, tamanho reduzido e baixo custo, recomendado para aplicações na internet das coisas. A plataforma utiliza-se de uma camada simples de software implementada na placa que utiliza a linguagem C/C++, a qual é *open source* também. Por ser uma plataforma de hardware livre existem diversos fabricantes desta e os mesmos utilizam nomes diferentes para suas placas, porém todas elas possuem a mesma finalidade.

3.6.1 Hardware

A placa utilizada no projeto será a ESP32 na versão ESP-WROOM-32. Esta placa possui um microcontrolador ESP32-D0WDQ6 de 12 bits e dois núcleos (dual core). As especificações da Tabela 2 estão de acordo com a documentação (ESPRESSIF, 2018c).

Tabela 2- Especificações do ESP-WROOM-32.

ESP-WROOM-32	
Tensão de Operação (Recomendada)	3,3 V
Memória interna ROM	448 KB
Memória interna SRAM	520 KB
Frequência de <i>clock</i> interno	40 Mhz
Corrente de operação (média)	80 mA

Fonte: Adaptado de ESPRESSIF (2018c).

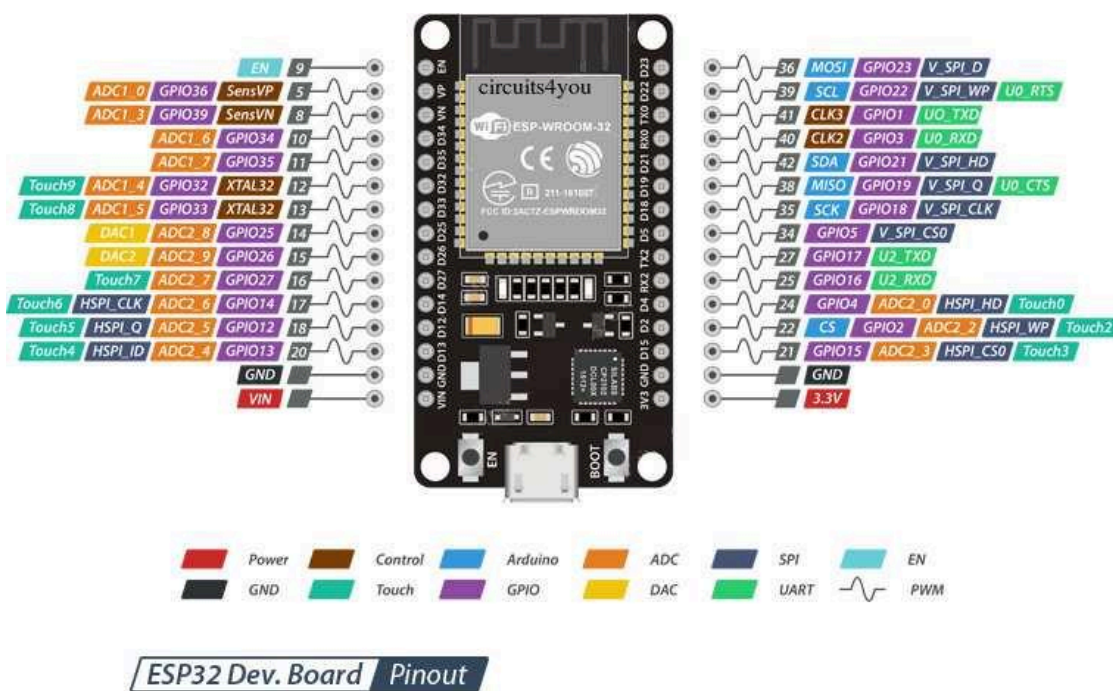
A placa do ESP32-WROOM-32 possui um total de 32 portas digitais, 16 delas podem ser utilizadas como saída PWM de 12bits. As portas digitais operam em uma tensão de 0V para nível lógico baixo e 3,3V para nível lógico alto, podendo fornecer uma corrente de 80mA. (ESPRESSIF, 2018c).

Possui um total de 18 entradas analógicas com resolução de 12bits para conversão digital. Estas portas de entrada são responsáveis pela aquisição de dados dos sensores que vem em formato analógico, quer são convertidos para uma escala digital com valores entre 0 a 4095 o que equivale aos 12bits, e estes valores posteriormente serão manipulados no software (ESPRESSIF, 2018b).

Os pinos de tensão existentes na placa de acordo com a documentação (ESPRESSIF, 2018c) são:

- Vin: Tensão de saída com valor de tensão igual à da fonte de alimentação da placa.
- 3,3V: Tensão de saída com 3,3V fornecida pela placa com corrente máxima de 80mA.
- GND: Pino conhecido como terra, que realiza fechamento do circuito.

Figura 5- Portas ESP32 WROOM32 DevKit.



Fonte: CIRCUITS4YOU, 2018.

Segundo (ESPRESSIF, 2018c) a placa possui dez pinos GPIO com suporte a toque capacitivo, dois canais conversores digital-analógico (DAC) de 8 bits, duas portas com interface I²C, muito utilizadas para módulos externos a placa, duas portas com interface UART, e duas portas com interface I²S, utilizadas para conectar componentes de áudio digital.

3.7 Sensores

Os sensores são responsáveis pela obtenção de valores de entrada no ESP32, com a aquisição dos valores é possível convertê-los para um formato digital em uma resolução de 12 bits através das portas analógicas e posteriormente manipular os mesmos dentro do software.

3.7.1 Tensão

Para realizar a medição da tensão será utilizado um divisor de tensão com resistores, cuja tensão de saída seja próxima aos valores máximos de leitura que o ESP32 suporta que é de 3,3V.

O sensor de tensão do projeto tem como objetivo identificar se há tensão após o disjuntor ou não para determinar se houve acionamento do mesmo, também é responsável pela medição da fase de entrada do quadro.

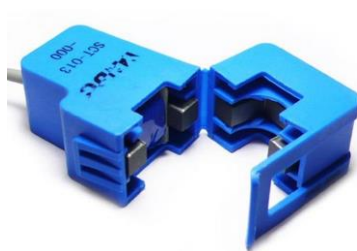
Os valores de tensão serão calculados por um algoritmo que irá adquirir amostras de tensão e fazer a integração dos mesmos retornando o valor RMS.

3.7.2 Corrente

O sensor de corrente utilizado nesse projeto será o não invasivo SCT-013-000, do inglês *Split core Current Transformer*, o funcionamento é idêntico a todo transformador de corrente, existem dois enrolamentos, sendo o primário o condutor de fase ou de neutro da instalação monitorada; e o secundário, alojado dentro do sensor, irá fornecer os sinais ao sistema. O seu funcionamento é baseado no princípio de indução magnética, onde a corrente elétrica que circula no enrolamento primário, induz uma corrente no enrolamento secundário, assim a segunda corrente é proporcional a primeira corrente (HUINFINITO, 2016).

O sensor SCT-013-000 pode transformar sinais de corrente alternada de grande amplitude em sinais de pequena amplitude. A corrente máxima que pode ser lida por ele é de 100 Ampères, podendo operar em temperaturas entre -25°C à 70°C , e seu enrolamento secundário possui 2000 espiras (HUINFINITO, 2016).

Figura 6- Sensor SCT-013-000.



Fonte: Huinfinito, 2016.

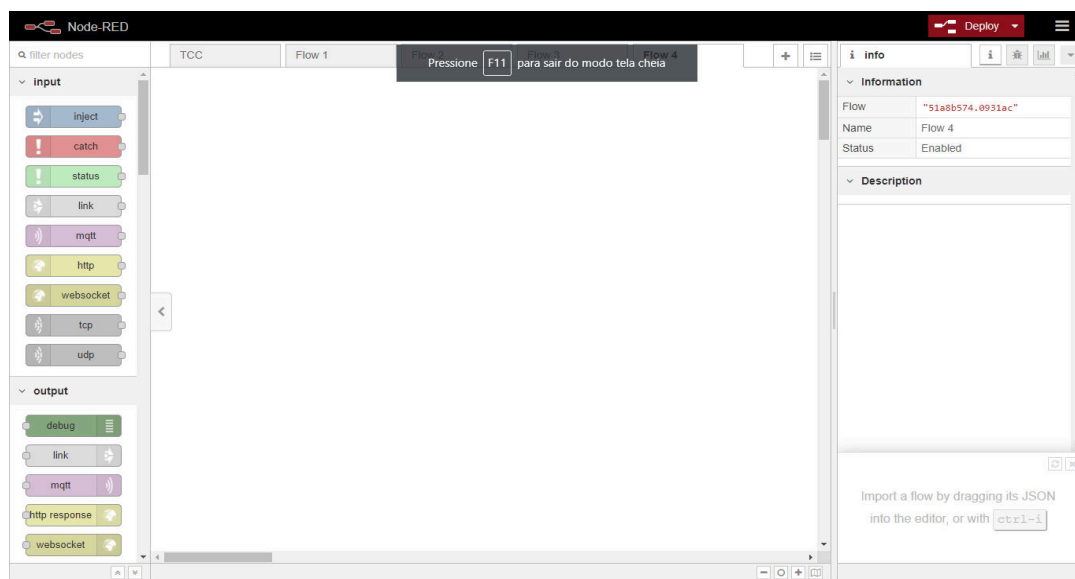
Este sensor foi escolhido para o projeto pelo fato do mesmo não ser invasivo, ou seja, a instalação do sensor é feita através da abertura do seu núcleo e então o mesmo é clipado no condutor escolhido, assim não é necessário seccionar o circuito.

Este sensor foi escolhido para o projeto pelo fato do mesmo não ser invasivo, ou seja, não há contato direto com a rede elétrica evitando possíveis danos à placa eletrônica ESP32. O mesmo é de fácil instalação e possui uma boa precisão.

3.8 Plataforma node-red

Node-Red é uma ferramenta de programação capaz de conectar dispositivos de *hardware*, APIs, e serviços online de maneira diferentes. Fornece um editor baseado em um navegador *web* que facilita o processo, possui um amplo diretório de nós que podem ser utilizados com um simples clique. A ferramenta é construída em Node.js que ideal para rodar em hardware de baixo custo e na nuvem (NODERED.ORG, 2019).

Figura 7- Interface da ferramenta Node-Red.



Fonte: autoria própria.

A Figura 7 mostra a interface da ferramenta com os seguintes diretórios: a esquerda está localizada todos os nós instalados, no centro o editor de *Flows* e a direita encontra-se abas de informações sobre os nós.

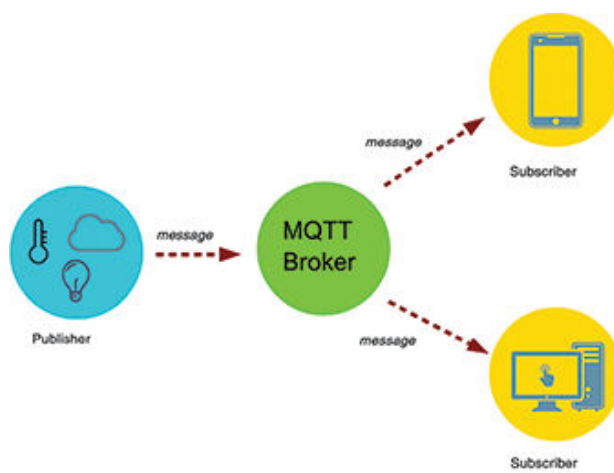
A ferramenta Node-Red, é utilizada neste projeto para conectar a plataforma com os dispositivos de hardware, a comunicação é feita pelo protocolo MQTT. A ferramenta também é capaz de processar as informações recebidas e enviar comando correspondentes a tais informações.

3.9 Protocolo mqtt

O principal protocolo utilizado para comunicação entre aplicações é o MQTT (*Message Queuing Telemetry Transport*). A utilização deste protocolo neste projeto se dá devido a sua praticidade de utilização e ser de fácil processamento.

O protocolo MQTT é baseado em mensagens leves e simples de *publish/subscribe* (publicar/ subscrever), seus princípios são em minimizar a largura de banda e os recursos dos dispositivos e também assegurando a confiabilidade e a segurança de comunicação. (MQTT.ORG, 2019).

Figura 8- Arquitetura de rede MQTT.



Fonte: Adaptado de

O modelo de publicar/subscrever usado pelo MQTT, possibilita a conexão de fontes de informação (publisher) com usuários (subscriber) por intermédio de um servidor (broker). As mensagens enviadas pelo publicante possuem o identificador de um tópico, assim o servidor distribui a mensagem para todos os dispositivos ou aplicações que estão inscritos naquele tópico (LAMPKIN et al., 2012).

4 MATERIAIS E METODOS

Neste capítulo serão abordados os procedimentos realizados para obter o protótipo final. Utilizando os sensores e componentes mostrados no capítulo anterior. Também são tratados aspectos importantes da programação em linguagem C utilizada para programar a placa eletrônica ESP32 e fazer com que esta envie os dados coletados para o banco de dados. Finalmente, como estes dados são tratados para que o usuário final tenha a informação da forma mais clara o possível.

4.1 Primeiros testes

4.1.1 Corrente

Sendo o sensor SCT-013-000 um transformador de corrente tem-se que as variações na sua saída são do tipo corrente alternada, no entanto, as entradas analógicas do ESP32 trabalham com sinais de amplitude de 0~3,3V não trabalhando com valores negativos de tensão, assim é necessário o condicionamento do sinal para que este atenda esta especificação.

Este condicionamento se dá através primeiramente de um offset no sinal do sensor utilizando uma tensão de 3,3V vinda próprio modulo ESP32, aplicado através de um divisor de tensão, com dois resistores de 47kΩ, de forma que este offset tem a função de deslocar a forma de onda senoidal do sinal do sensor para impedir a existência de valores negativos.

Sabendo que o sensor de corrente tem uma variação de corrente de 0 à 50mA na saída, se faz necessário a adição de um resistor de carga para que seja gerada uma variação de tensão que vai ser lida pelo modulo ESP32. Para a escolha do resistor de carga foi realizado o seguinte cálculo:

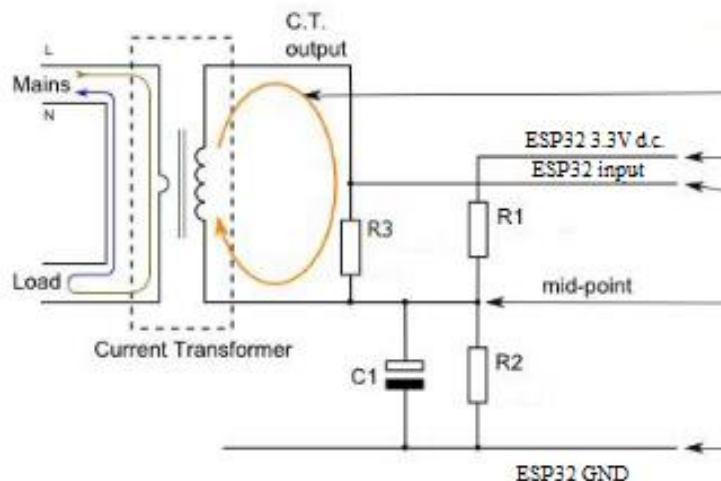
$$i_{sensor} = \frac{i_{medido}}{n_{espiras}} \quad (18)$$

$$i_{sensor} = \frac{25A \cdot \sqrt{2}}{2000} = 0,01777A \quad (19)$$

$$R_{carga} = \frac{V_{sensor}}{I_{sensor}} = \frac{1,65V}{0,01777A} = 92,85\Omega \quad (20)$$

O resistor de carga R3 escolhido foi de 82Ω valor comercial atendendo o requisito de não ultrapassar o valor 3,3V, não danificando o microcontrolador. Já os valores das resistências R1 e R2 foram de $47k\Omega$.

Figura 9-Esquema de ligação para o TC.



Fonte: Adaptado OPEN ENERGY MONITOR, 2019.

Por fim foi adicionado um capacitor de $10\mu F$ para fornecer um caminho de baixo impedância para o sinal CA, diminuindo assim o ruído de alta frequência.

4.1.2 Tensão

Outro teste realizado foi da aquisição de tensão através de divisor de tensão, dispensando a utilização de transformadores pois apresenta dimensões elevadas. A metodologia de inserção de uma Vcc para que não haja leitura negativa de tensão segue a mesma lógica que a medição de corrente.

4.2 Programação do microcontrolador

O ambiente de programação escolhido para este projeto foi o Arduino IDE, pois este é compatível com a placa eletrônica ESP32 a ser utilizada, e possui bibliotecas como a *Electricity monitoring library* (Emonlib), facilitando assim a obtenção dos dados desejados.

4.2.1 Código da biblioteca

A biblioteca de monitoramento de eletricidade (emonlib), é uma biblioteca projetada para o Arduino com comandos de cálculo e monitoramento de grandezas elétricas como tensão, corrente e potências, fornecida gratuitamente pela comunidade *OpenEnergyMonitor* (OPEN ENERGY MONITOR, 2019).

Como esta biblioteca foi desenvolvida para as placas Arduino que possuem uma tensão de entrada nas portas ADC entre 0 e 5V com uma resolução de 10bits, se fez necessário realizar testes práticos para adaptar a biblioteca para a configuração do ESP32 que possui ADC entre 0 e 3,3V e resolução 12bits.

Foi realizado um experimento para calibração do mesmo utilizando um exemplo da biblioteca emonlib, para evitar distorções ao resultado das medições. As constantes adotadas foram práticas, ou seja, foram modificadas até os valores do monitor serial da IDE Arduino e dos instrumentos de medição entrarem em paridade. A Figura 10 ilustra o código utilizado para calibração.

Figura 10- Código de calibração utilizado no projeto.

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

#include "EmonLib.h"          // Include Emon Library
EnergyMonitor emon1;         // Create an instance
#define ADC_BITS      12
#define ADC_COUNTS    (1<<ADC_BITS)

void setup()
{
  Serial.begin(9600);

  emon1.voltage(34, 77.20, 24); // Voltage: input pin, calibration, phase_shift
  emon1.current(36, 5.7);      // Current: input pin, calibration.
}

void loop()
{
  emon1.calcVI(20,2000);      // Calculate all. No.of half wavelengths (crossings), time-out
  emon1.serialprint();        // Print out all variables (realpower, apparent power, Vrms, Irms, power factor)

  float realPower    = emon1.realPower;    //extract Real Power into variable
  float apparentPower = emon1.apparentPower; //extract Apparent Power into variable
  float powerFactor   = emon1.powerFactor;  //extract Power Factor into Variable
  float supplyVoltage = emon1.Vrms;         //extract Vrms into Variable
  float Irms          = emon1.Irms;         //extract Irms into Variable
}
```

Fonte: Autoria própria

O método de calibração utilizado foi o de comparação, os valores obtidos dos sensores eram comparados com o do alicate amperímetro Minipa ET-3111 que servia de referência para os valores de tensão eficaz e corrente eficaz. O mesmo está aferido, apresentando uma precisão do para tensão eficaz é $\pm 1.5\%$ com uma resolução de 1V, e

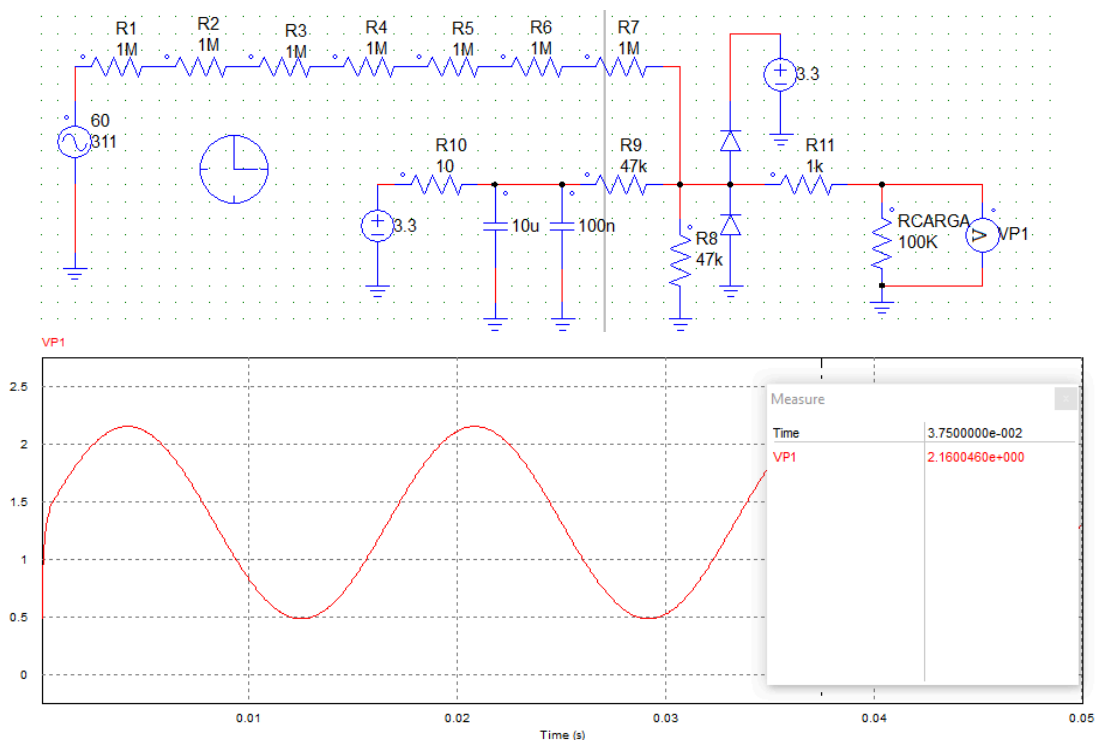
para corrente eficaz a precisão é de $\pm 3\%$ com uma resolução de 10mA. Assim, o mesmo apresenta uma precisão aceitável para a calibração.

Depois dos testes realizados, foi constatado que o valor de calibração utilizado no código neste projeto é de 6 para a corrente, 77,20 para valores de tensão e 24 para os valores de fase.

4.3 Placa

Para o desenvolvimento da placa para este projeto levou-se em conta que o aparelho estará diretamente conectado à rede para realizar a leitura da tensão e indiretamente para realizar a leitura da corrente. Como estamos trabalhando com tensão alternada na faixa dos 220Vrms, como o ESP32 não trabalha com tensões alternadas dessa amplitude, foi necessário reduzir a tensão para satisfazer as condições de leitura para o mesmo. Para isto foram escolhidos componentes eletrônicos a fim de reduzir a tensão para uma faixa aceitável, limitar surtos de tensão e proteger a porta AD, afim de evitar danificar as portas do ESP32.

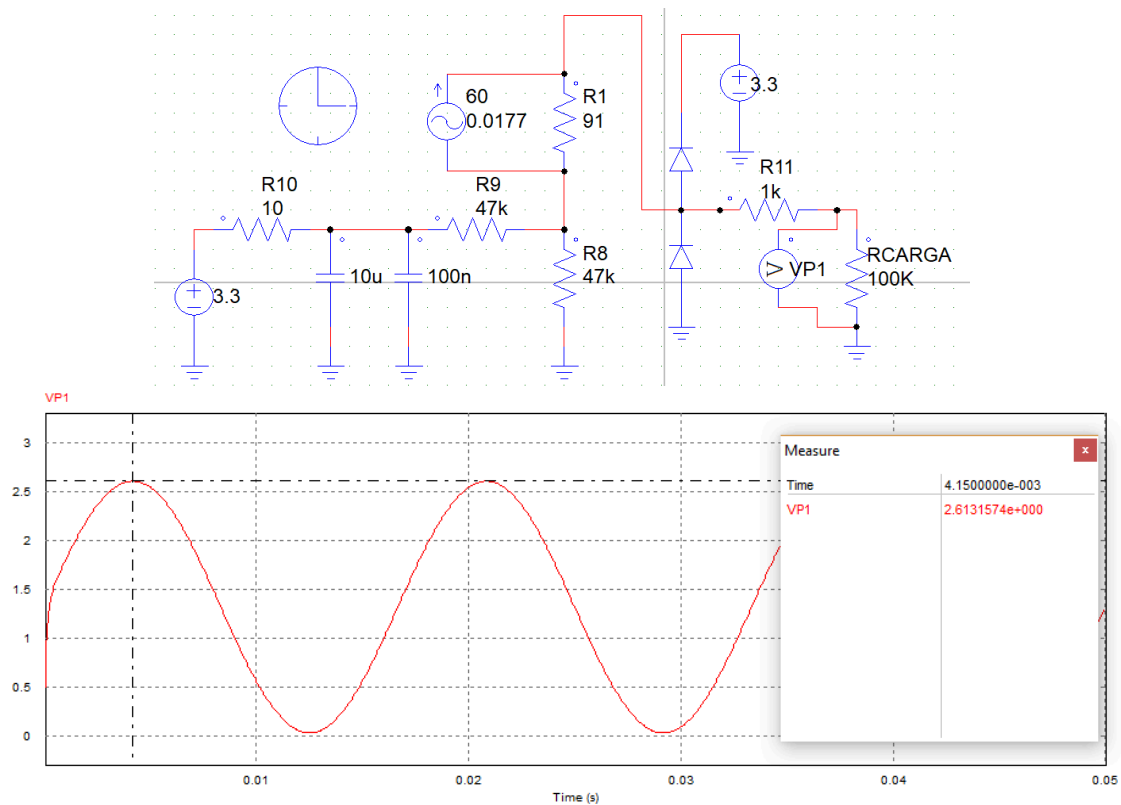
Figura 11- Circuito de tensão simulado no PSIM.



Fonte: Autoria própria.

A Figura 11 ilustra simulação realizada no PSIM do sensor de tensão com a proteção para a porta AD , e verificação da forma onda e sua faixa de tensão na entrada do ESP32.

Figura 12-Circuito de corrente simulado no PSIM.

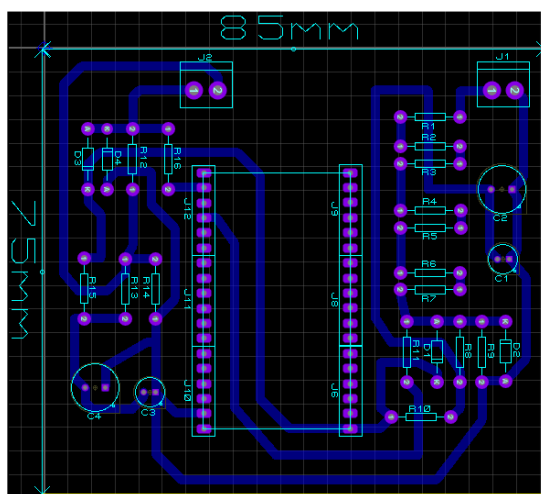


Fonte: Autoria própria.

A Figura 12 mostra simulação realizada no PSIM do sensor de corrente com a proteção de polarização reversa e verificação da forma onda e sua faixa de tensão na entrada do ESP32.

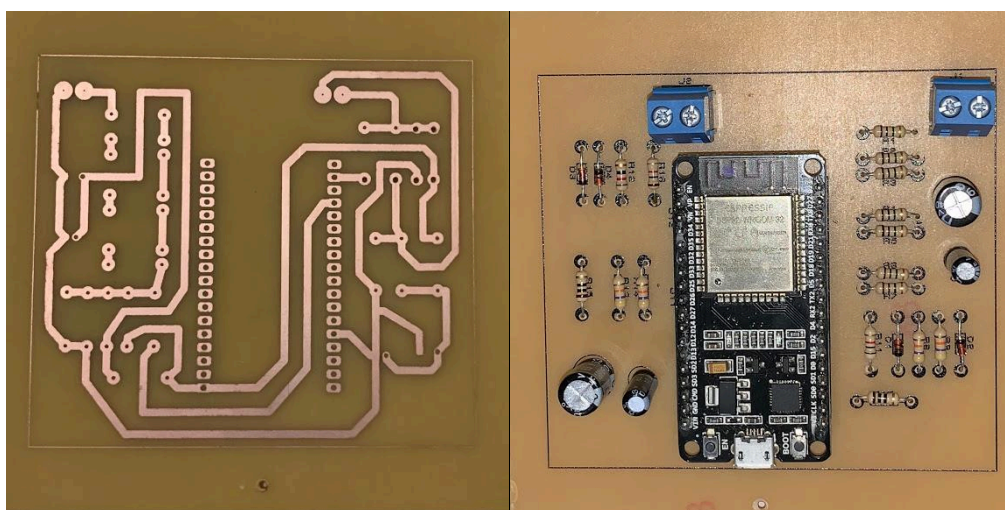
Finalizada as simulações e com todos os componentes eletrônicos definidos iniciou a confecção da PCB utilizando o software Proteus. O circuito dos dois sensores se encontram nas duas extremidades separados pelo ESP32. A PCB foi confeccionada em uma placa de fenolite cobreada e corroída com percloroeto de ferro. O resultado está ilustrado na Figura 14.

Figura 13- Placa desenvolvida no software Proteus.



Fonte: Autoria própria.

Figura 14- Placa corroída e com seu componentes fixados.



Fonte: Autoria própria.

Na Figura 13- Placa desenvolvida no software Proteus. Figura 13 ilustra a placa desenvolvida no software Proteus e já a Figura 14 ilustra a placa de fenolite corroída a esquerda e a mesma com todos os componentes eletrônicos soldados em seus devidos locais a direita. Finalizando o protótipo de supervisão.

4.4 Manipulação de dados

O projeto tem como objetivo facilitar a detecção de cargas que estão utilizando a rede. Esta facilidade se dá devido ao usuário acessar as informações em tempo real e de

forma acumulada essas exibidas em gráficos. O acesso pode ser feito por qualquer equipamento que esteja conectado à rede local, seja ele celular, tablete ou computador. Para visualização destas informações basta acessar um navegador *web*.

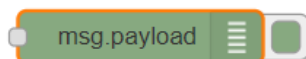
O Node-Red foi instalado a um computador, que terá função de servidor. Assim, responsável por receber todo o fluxo de dados da rede local onde o ESP32 se encontra conectado.

Todas as informações processadas pelo ESP32, obtidas através da leitura dos sensores, serão publicadas em tópicos na rede local. Com o Node-Red inscrito nestes tópicos é possível receber os dados presente na rede, processa-los, armazena-los e por fim exibi-los em um navegador *web*.

Neste projeto foram utilizando os seguintes nós para configurar o recebimento, armazenamento e exibição dos dados.

O nó **debug**: Exibe as propriedades de mensagens selecionadas na guia da barra lateral de depuração e, opcionalmente, no log de tempo de execução; ele pode ser adicionado e conectado a todos os demais nós do Node-Red para acompanhamento e identificação de problemas.

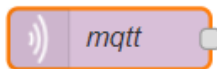
Figura 15- Nó de debug.



Fonte: Autoria própria.

O nó **mqtt**: Possui duas variações, uma publicação e outra de inscrição. A utilizada é a de publicação, que se configura o tópico do MQTT do servidor publicando os dados recebidos.

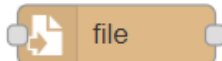
Figura 16- Nó MQTT.



Fonte: Autoria própria.

O nó **file**: Grava msg.payload em um arquivo, seja adicionando ao final ou substituindo o conteúdo existente. O formato de arquivo escolhido para este projeto foi o csv.

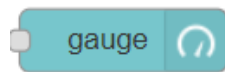
Figura 17- Nó file.



Fonte: Autoria própria.

O nó **gauge**: Adiciona um widget de tipo de medidor à interface do usuário.

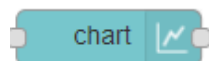
Figura 18- Nó gauge.



Fonte: Autoria própria.

O nó **chart**: Plota os valores de entrada em um gráfico. Isso pode ser um gráfico de linhas com base no tempo, um gráfico de barras (vertical ou horizontal) ou um gráfico de pizza.

Figura 19- Nó chart.

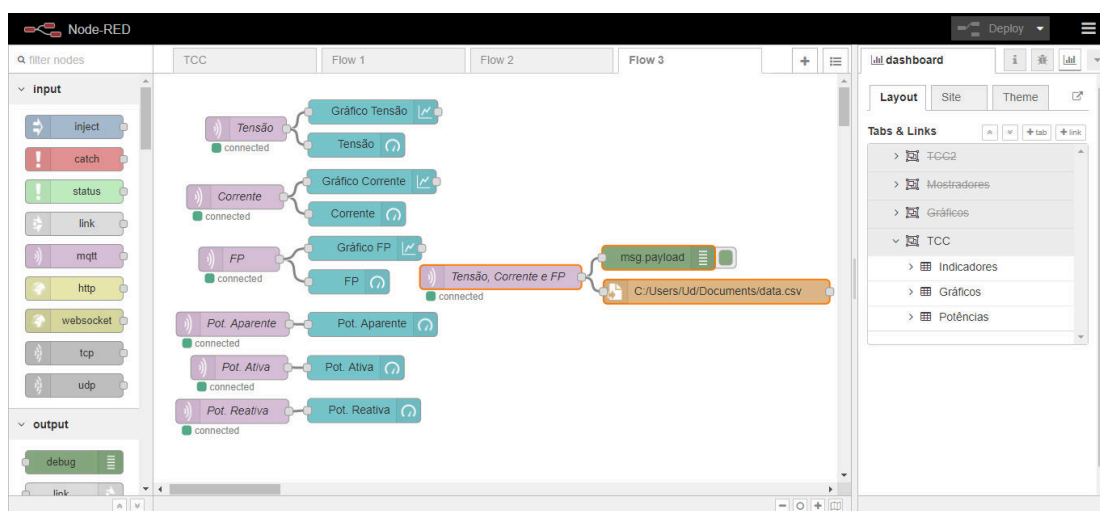


Fonte: Autoria própria.

Na Figura 20- ilustra o *flow* utilizado no projeto, inicialmente as informações chegam a plataforma Node-Red por nós de entrada de protocolo MQTT, o nó “Tensão” recebe as informações da leitura do sensor de tensão; o nó “Corrente” as informações da leitura do sensor de corrente; O restante dos nós MQTT (FP, Pot. Aparente, Pot. Ativa, Pot. Reativa) recebe as informações referente aos cálculos realizados pelo ESP32 para obtenção destes valores; O nó “Tensão, Corrente e FP” recebe uma *string* com as informações referentes ao título do nó separados por vírgula.

Todos os nós MQTT estão conectados ao nó *gauge*. Apenas os nós “Tensão”, “Corrente” e “FP” estão conectados ao nó *chart*. E o único que está conectado aos nós *debug* e o *file* é o nó “Tensão, Corrente e FP”.

Figura 20- Página web de edição.



Fonte: Autoria própria.

O nó *gauge* é responsável por adiciona um *widget* de tipo de medidor para acompanhamento em tempo real dos valores recebidos, o nó *chart* responsável por plotar os valores recebidos em um gráfico para o acompanhamento em um determinado

intervalo de tempo. Já o nó *debug* é responsável por acompanhar o recebimento de dados sem necessidade de abrir a interface do sistema supervisorio, por fim o nó *file*, responsável por salvar as leituras em um arquivo .csv, afim de utilizar outros programas para realizar um estudo de consumo elétrico mais detalhado.

Figura 21- Interface *web* do Node-Red.



Fonte: Autoria própria.

Na Figura 21- temos a interface *web* do Node-Red que é acessado pelo endereço “<http://127.0.0.1:1880/ui/#!/0>”, ele mostra o funcionamento do sistema, disponibilizando os valores de cada componente nas *widget*.

5 RESULTADO E DISCUSSÕES

Este capítulo tem como objetivo apresentar os resultados obtidos em alguns testes realizados após a montagem do medidor. Testes realizados são apresentados juntamente com as divergências encontradas no comparativo (dados de placa e aparelhos de medição). Por fim os resultados obtidos são analisados e discutidos em respeito à precisão do equipamento.

5.1 Teste com lâmpada incandescente 150w

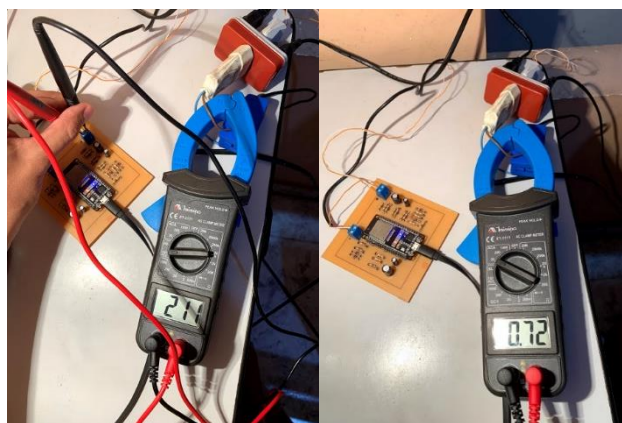
Para o primeiro teste realizado com o protótipo do medidor após a montagem, foi utilizado uma lâmpada incandescente 150 Watts. Pois, segundo Freitas (2016) o teste com esse tipo de carga tem o objetivo de avaliar a precisão do medidor construído com uma carga linear, bem como validar a diferença entre os tipos de carga enfatizando a relação do triângulo de potências.

O esperado para este tipo de carga é um fator de potência próximo ao unitário, assim, sua potência ativa e aparente devem ter valores próximos e uma baixa potência reativa.

O equipamento utilizado para comparação é um alicate amperímetro digital da Minipa, este segundo seu manual possui uma resolução para corrente de 10mA e uma precisão de 3%, já para tensão possui uma resolução de 1V e uma precisão de 1,5%. Assim os valores de tensão e corrente medidos pelo alicate amperímetro para a lâmpada de 150W estão representados na

Figura 22.

Figura 22- Medição da tensão e corrente da lâmpada de 150W com o alicate amperímetro.



Fonte: Autoria própria.

Tabela 3- Valores de tensão e corrente lidos pelo alicate no experimento 5.1

Tensão	211V
Correntes	0,72A

A Figura 23 ilustra a captura de tela da interface do sistema supervisor no instante em que as medidas com o alicate amperímetro estavam sendo realizadas.

Figura 23-Amostra da interface do sistema supervisor com valores equivalente do experimento 1.



Fonte: Autoria própria.

Para calcular o erro absoluto foram utilizados os valores alicate amperímetro como reais, o erro do equipamento foi então calculado em comparação com estes valores. Para a corrente, temos o exemplo de uma corrente real de 0,72A, e o valor medido de 0,72A, que nos leva a um erro absoluto que pode ser calculado pela Equação (21).

$$E\% = \frac{|Vm - Vr|}{Vr} \times 100\% \quad (21)$$

Onde: Vm: Valor medido pelo programa. Vr: Valor real, medido pelo amperímetro alicate. Logo, chegamos ao valor de erro absoluto igual a:

$$E\% = \frac{|0,72 - 0,72|}{0,72} * 100\% = 0,00\% \quad (22)$$

O mesmo pode ser feito para calcular o erro absoluto da medição de tensão, desta vez comparando os valores encontrados pelo programa com estes medidos pelo multímetro. Para tensão real temos 211V, e valor medido de 211,18V. O que resulta no valor do erro absoluto de tensão aproximado a 0,1%.

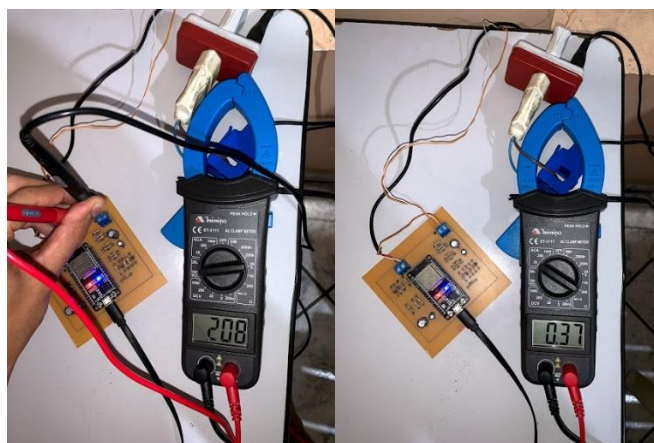
Ao final do teste foram obtidos valores muito próximos dos esperados, assim pôde-se averiguar que as potências aparente e ativa tiveram valores próximos, sendo estes também próximos ao valor nominal da lâmpada, assim como esperado. Constatou-se ainda que o erro percentual ficou abaixo de 0,2%.

5.2 Teste com ventilador 80w

Para o segundo teste realizado com o protótipo foi utilizado um ventilador ARNO cuja a potência de nominal informada nas especificações técnicas do ventilador é de 80W em sua velocidade máxima.

Com o ventilador ligado em sua menor velocidade iniciou-se a medição com o alicate amperímetro para obter os valores de tensão e corrente. A Figura 24 ilustra esses valores medidos.

Figura 24- Medição da tensão e corrente do ventilador com o alicate amperímetro



Fonte: Autoria própria.

Tabela 4-Valores de tensão e corrente lidos pelo alicate no experimento 5.2

Tensão	208V
Correntes	0,37A

A

Figura 25 ilustra a captura de tela da interface do sistema supervisorio no instante em que as medidas elétricas do ventilador estavam sendo realizadas com o alicate amperímetro.

Figura 25-Amostra da interface do sistema supervisório com valores equivalente do experimento



Fonte: Autoria própria.

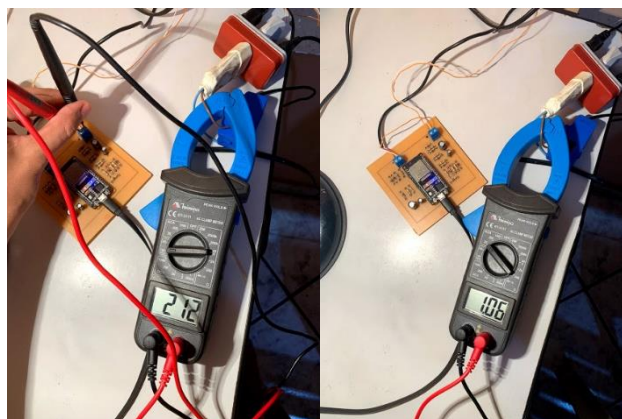
Para calcular o erro absoluto da corrente temos uma corrente real de 0,37A, e o valor medido de 0,38A. O que resulta no valor do erro absoluto de corrente foi de aproximadamente 2,8%. Já para o erro absoluto da tensão temos um valor real de 208V, e um valor medido de 207,86V. Resultando um erro inferior a 0,1% para a tensão.

Com este teste foi possível observar que para cargas de corrente elétrica pequenas o sistema não apresenta uma boa precisão, mas para o fim de detecção de cargas na rede o mesmo é satisfatório.

5.3 Teste com lâmpada incandescente 150w e ventilador de 80w

Para o terceiro teste foi utilizado as duas cargas dos testes anteriores, afim de querer elevar a potência a ser medida e vê o comportamento do fator de potência na interfasse ao utilizar uma carga resistiva e outra indutiva.

Figura 26- Medição da tensão e corrente da lâmpada incandescente e ventilador com o alicate amperímetro.



Fonte: Autoria própria.

Tabela 5-Valores de tensão e corrente lidos pelo alicate no experimento 5.3

Tensão	212V
Correntes	1,06A

A Figura 26 ilustra o alicate amperímetro medindo tensão e corrente da carga do terceiro teste. O valor lido de tensão foi 212V e para corrente 1,06A.

Figura 27- Amostra da interface do sistema supervisor com valores equivalente do experimento 3.



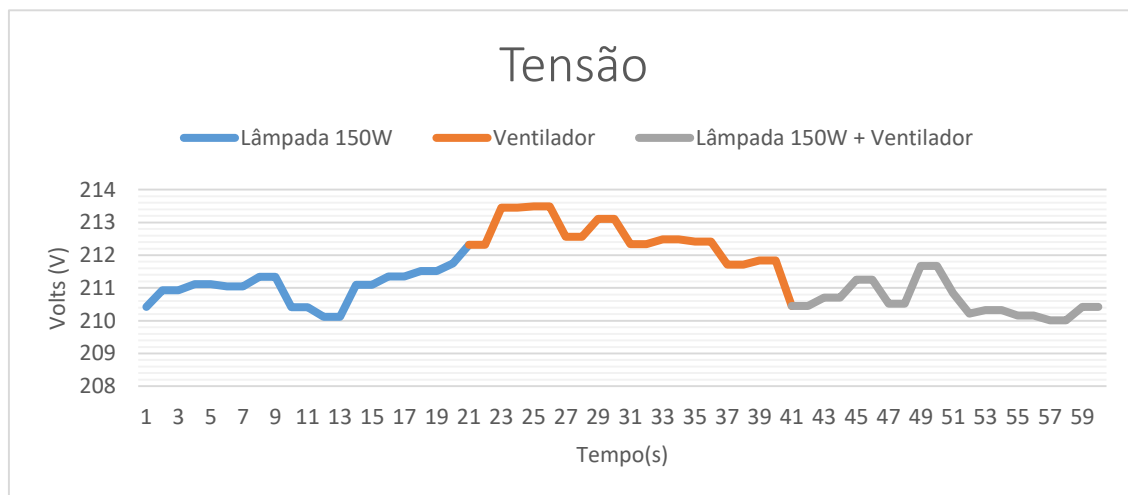
Fonte: Autoria própria.

Conforme a captura de tela ilustrada na Figura 27, observou-se que a corrente informada na interface é o equivalente a soma das duas correntes medidas pelo alicate amperímetro dos testes anteriores. Assim, confirmando que o protótipo é capaz de detectar utilização das duas cargas no mesmo intervalo de tempo.

5.4 Teste de variação de cargas

Os testes citados acima foram realizados novamente em um intervalo de tempo menor, afim de conseguir capturar essas variações. Os dados utilizados para os gráficos Gráfico 1, Gráfico 2 e Gráfico 3 foram os do arquivo data.csv, estes armazenados pelo sistema de supervisão desenvolvido.

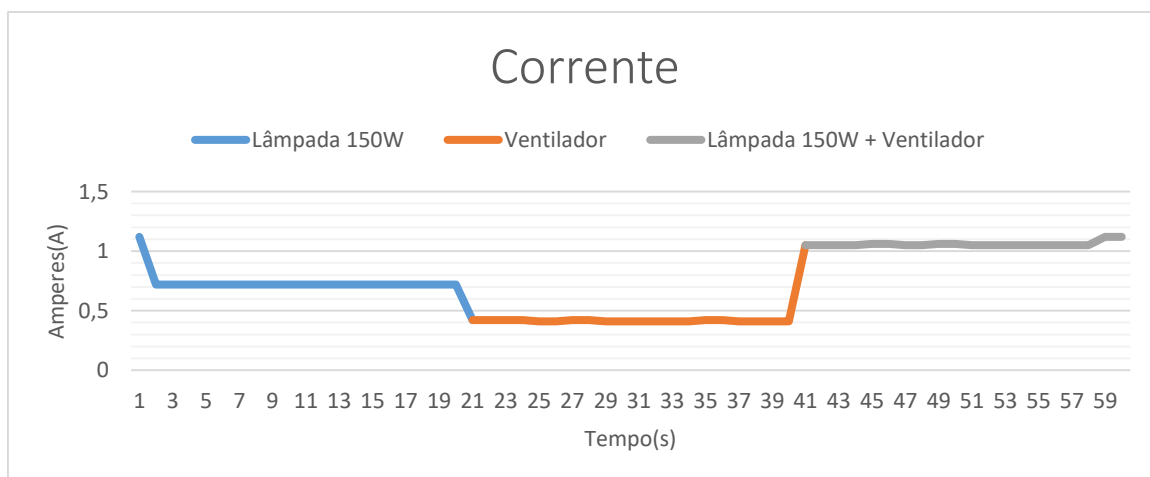
Gráfico 1- Variação da tensão ao aplicar diferentes cargas.



Fonte: Autoria própria.

O Gráfico 1 mostra o monitoramento da tensão durante o experimento, e foi possível observar a oscilação da tensão durante a troca de cargas. Quanto maior a potência da carga conectada à rede maior a queda de tensão.

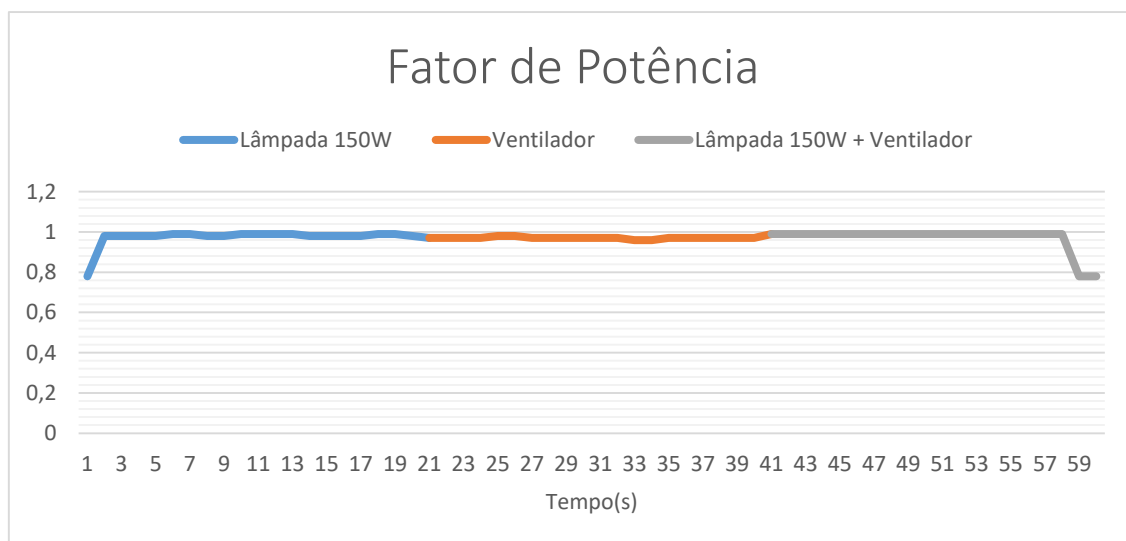
Gráfico 2- Variação da corrente ao aplicar diferentes cargas.



Fonte: Autoria própria.

O Gráfico 2 mostra monitoramento da corrente durante o experimento, assim, foi possível observar sua variação durante a troca de cargas. Apresentando valores mais elevados nos intervalos em que as cargas de maior potência foram adicionadas.

Gráfico 3- Variação da fator de potência ao aplicar diferentes cargas.



Fonte: Autoria própria.

O Gráfico 3 mostra monitoramento do fator de potência durante o experimento, assim, foi possível observar uma pequena variação quando o ventilador entrou tornando o sistema levemente indutivo.

6 CONCLUSÃO

O trabalho desenvolvido obteve resultados satisfatórios, pois conforme os resultados apresentados as medições estão próximas em comparação ao alicate amperímetro que foi utilizado como referência de medição e calibração. Porém não é possível afirmar se dados obtidos ao longo de um período maior, um mês por exemplo, serão compatíveis com o consumo real. No entanto a finalidade deste medidor não é de se obter o consumo real predial ao longo do mês. A princípio o objetivo é quantificar e determinar onde está sendo consumida a energia elétrica do prédio através dos sensores ligados aos ramais de derivação do quadro de distribuição elétrico, e com esses valores encontrar maneiras para limitar o uso da energia, tornando mais sustentável.

O experimento onde foi utilizada com uma carga puramente resistiva e os valores lidos pelo sensor e apresentados na interface mostraram-se compatíveis com os valores medidos pelo alicate amperímetro. Nas medições de corrente eficaz o dispositivo, apresentou uma maior variação quando comparado as medições de tensão elétrica, evidenciando que para cargas com correntes pequenas estas acentuam-se.

Tais variações devem-se além de oscilações na rede ao fato que a medição é feita de forma indireta, pois o sensor faz uma leitura de variação de corrente, porém o ESP32 lê variações de tensão. Mesmo com estas variações as leituras apresentaram discrepâncias máxima de 0,06A para cargas muito pequenas. Para a tensão as medições sofreram uma menor variação, sendo estas causadas devido a oscilações na rede, apresentando um erro máximo de 0,1% com relação as medições realizadas utilizando o alicate amperímetro de referência.

Por fim, o sistema de monitoramento apresentado aplicado a instalação de forma geral apresentou dados que podem ser de grande valia. A partir do momento em que o consumidor passa a ter um controle sobre seu consumo, com as informações oferecidas pelo sistema, passa-se a ter uma visão mais ampla sobre as possibilidades de economia de energia.

7 REFERENCIAS

ALEXANDER, Charles K. SADIKU, Matthew N. O. **Fundamentos de circuitos elétricos**. Porto Alegre – RS: AMGH, 2013. 874p.

CIRCUITS4YOU, ESP32 Devkit ESP32-WROOM GPIO Pinout. 2018. Disponível em: <<https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout/>>. Acesso em: 11/07/2019.

DATAWEEK. Quectel suporta protocolo MQTT para IoT. 2018. Disponível em: <<https://www.dataweek.co.za/9101a>>. Acesso em: 03/08/2019.

FREITAS, Elton da Silva. Sistema de aquisição de dados para monitoramento do consumo de energia elétrica de uma unidade consumidora residencial, Monografia (Bacharelado em Ciência e Tecnologia). Universidade Federal Rural do Semi Árido. Mossoró, 2016.

HUINFINITO, Sensor de corrente não invasivo SCT013-100A, 2016. Disponível em <<http://www.huinfinito.com.br/sensores/956-sensor-de-corrente-nao-invasivo-sct013100a.html>> Acesso em 20/07/2019.

JEGUNDO, Rafael. Development and monetization of an energy data and analytics platform, Dissertação (Master of Science in Physics Engineering). Faculty of Science and Technology of the University of Coimbra. Coimbra, 2012. Disponível em <https://estudogeral.sib.uc.pt/bitstream/10316/25202/1/rafael_jegundo.pdf> Acesso em 20/07/2019.

KUP, Mariana Torres. Estudo da Medição Inteligente para consumidores residenciais no Brasil. UFRJ, Rio de Janeiro, 2015. Disponível em: <<http://monografias.poli.ufrj.br/monografias/monopoli10014242.pdf>>. Acesso em: 17/07/2019.

LAMPKIN, V. et al. Building Smarter Planet Solutions with MQTT and IBM WebSphere MQ Telemetry. 2012. Redbook da IBM sobre MQTT. Disponível em: <<http://www.redbooks.ibm.com/redbooks/pdfs/sg248054.pdf>>. Acesso em: 22/07/2019.

MARTINEZ, Pedro C. S. Wattímetro, Relatório Final da Disciplina de Instrumentação Eletrônica. Universidade Católica de Pelotas. Pelotas, 2013. Disponível em: <http://olaria.ucpel.tche.br/ie2013/lib/exe/fetch.php?media=relatorio_final_wattimetro.pdf> Acesso em 25/07/2019.

MINHA CASA SOLAR. **Energia solar**. 2019. Disponível em: <http://blog.minhacasasolar.com.br/smart-grid/>>. Acesso em: 22/07/2019.

MORO, Jefferson Zortéa. Medidor de consumo de energia elétrica descentralizado e com interface web. Universidade Estadual de Campinas. Campinas, 2013

MQTT.ORG. **Frequently Asked Questions**. 2019. Disponível em: <<http://mqtt.org/faq>>. Acesso em: 22/07/2019.

NANSEN. **Medidores**. 2019. Disponível em: <<http://nansen.com.br/medidores/lumen3/>> Acesso em: 24/07/2019.

NBR 8377. **Medidor de energia ativa – especificação**. 1995. Disponível em: <https://www.target.com.br/produtos/normas-tecnicas/29614/nbr8377-medidor-de-energia-ativa>>. Acesso em: 19/07/2019.

NODERED.ORG. **Flow-base programming for the Internet of Things**. 2019. Disponível em: <<https://nodered.org/#features>>. Acesso em: 22/07/2019.

OPEN ENERGY MONITOR. Electricity monitoring, 2019. Disponível em: <<https://learn.openenergymonitor.org/electricity-monitoring/ac-power-theory/introduction>>. Acesso em: 20/07/2019.

UFJF NOTÍCIAS. Redes elétricas inteligentes já são realidade; entenda os desafios da nova tecnologia. 2017. Disponível em: <<https://www2.ufjf.br/noticias/2017/04/17/redes-eletricas-inteligentes-ja-sao-realidade-entenda-os-desafios-da-nova-tecnologia/>>. Acesso em: 18/07/2019.

VELASCO, Loana Nunes. Análise experimental de erros de medição de energia elétrica ativa em medidores eletromagnéticos tipo indução, sujeitos a distorções harmônicas de correntes e tensões, em sistemas equilibrados e desequilibrados. São Paulo, 2007. 187 f. Disponível em: <<https://repositorio.unesp.br/handle/11449/87277>>. Acesso em: 20/07/2019.

VAGAPOV, Yuriy. Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things. 2017. Disponível em: <https://www.researchgate.net/publication/320273388_Comparative_Analysis_and_Practical_Implementation_of_the_ESP32_Microcontroller_Module_for_the_Internet_of_Things>. Acesso em: 21/07/2019.

8 APÊNDICE A – CÓDIGO FINAL DO SISTEMA SUPERVISÓRIO

```
// -- Bibliotecas Auxiliares --
#include <WiFi.h>
#include <PubSubClient.h>
#include "EmonLib.h"

#define ADC_BITS 12
#define ADC_COUNTS (1<<ADC_BITS)
const int lamp = 2;

// -- Definições de Rede e conexão MQTT / Tópicos --
const char* ssid = "casa";
const char* password = "1010101010";
const char* mqtt_server = "10.215.115.173";

// -- Instância objetos --
// Inicializa o espClient. Você deve alterar o nome do espClient se tiver vários ESPs em
// execução no sistema de automação residencial
WiFiClient espClient;
PubSubClient client(espClient);
EnergyMonitor SCT013;

// -- Variáveis e constantes --
long now = millis();
long lastMeasure = 0;

// Não mude a função abaixo. Esta função liga o seu ESP32 ao seu router
void setup_wifi() {
  delay(10);
  // Começamos conectando a uma rede WiF
  Serial.println();
  Serial.print("Conectando a rede: ");
  Serial.println(ssid);
```

```

WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.print("WiFi Conectado - endereço IP ESP32: ");
Serial.println(WiFi.localIP());
}

// Esta função reconecta seu ESP32 ao seu corretor MQTT
// Altere a função abaixo se quiser se inscrever em mais tópicos com o seu ESP
void reconnect() {
  // Faça um loop até nos reconectarmos
  while (!client.connected()) {
    Serial.print("Conectando ao broker MQTT:");
    // Tentativa de conectar

    if (client.connect("ESP32Client")) {
      Serial.println(" conectado");
      digitalWrite(lamp, HIGH);

    } else {
      Serial.print("Falha estado: ");
      Serial.print(client.state());
      digitalWrite(lamp, LOW);
      Serial.println(" Tentar novamente em 5s");
      //Aguarde 5 segundos antes de tentar novamente
      delay(5000);
    }
  }
}

```

// A função de configuração define seus GPIOs de ESP para Saídas, inicia a comunicação serial a uma taxa de transmissão de 115200

// Configura seu broker mqtt e configura a função callback

```
void setup() {
  pinMode(lamp, OUTPUT);
  Serial.begin(115200);
  setup_wifi();
  client.setServer(mqtt_server, 1883);
```

```
SCT013.voltage(34, 77.20, 24); // Voltage: input pin, calibration, phase_shift
```

```
SCT013.current(36, 6);
```

```
}
```

// Para este projeto, você não precisa alterar nada na função de loop. Basicamente, garante que você está conectado ao seu corretor ESP

```
void loop() {

  if (!client.connected()) {
    reconnect();
  }
  if(!client.loop())
    client.connect("ESP32Client");
```

```
    SCT013.calcVI(120,2000);      // Calculate all. No.of half wavelengths (crossings),
time-out
```

```
    // SCT013.serialprint();      // Print out all variables (realpower, apparent power,
Vrms, Irms, power factor)
```

```
float supplyVoltage = SCT013.Vrms;      //extract Vrms into Variable
```

```
float Irms          = SCT013.Irms;      //extract Irms into Variable
```

```
float realPower     = SCT013.realPower;  //extract Real Power into variable
```

```
float apparentPower = SCT013.apparentPower; //extract Apparent Power into
variable
```

```
float powerFactor   = SCT013.powerFactor; //extract Power Factor into Variable
```

```
float potenciareativa = sqrt(pow(apparentPower,2) - pow(realPower,2)); //Retorna o
valor da potencia reativa
```

```
static char tensaoV[7];
dtostrf(supplyVoltage, 6, 2, tensaoV);
```

```
static char correnteA[7];
dtostrf(Irms, 6, 2, correnteA);
```

```
static char potAp[7];
dtostrf(apparentPower, 6, 2, potAp);
```

```
static char potAt[7];
dtostrf(realPower, 6, 2, potAt);
```

```
static char FP[7];
dtostrf(powerFACTOR, 6, 2, FP);
```

```
static char potrRea[7];
dtostrf(potenciareativa, 6, 2, potrRea);
```

```
char saveinf[24];
sprintf(saveinf, "%s,%s,%s", tensaoV, correnteA, FP);
```

```
// Publica valores
client.publish("esp32/savei", saveinf);
client.publish("esp32/tensao", tensaoV);
client.publish("esp32/corrente", correnteA);
client.publish("esp32/paparente", potAp);
client.publish("esp32/pativa", potAt);
client.publish("esp32/fatorpot", FP);
client.publish("esp32/preativa", potrRea);
}
```