

Desenvolvimento de um sistema embarcado para monitoramento de energia elétrica

João Pedro Almeida Pinto
Universidade Federal de Viçosa
Departamento de Engenharia Elétrica
Viçosa, Brasil
joao.pinto5@ufv.br

André Gomes Torres
Universidade Federal de Viçosa
Departamento de Engenharia Elétrica
Viçosa, Brasil
angoto@ufv.br

Resumo—O presente trabalho apresenta o desenvolvimento de um sistema embarcado, focado na telemetria e análise de energia elétrica, a partir do monitoramento de tensão, corrente, potência, consumo e fator de potência de um sistema. O projeto é baseado em um programa feito a partir da linguagem de programação Arduino, funcionando de forma dedicada em um microcontrolador com conexão Wi-Fi, com sinais de entrada captados por sensores analógicos. Os dados adquiridos são processados pelo código do programa e transferidos para uma plataforma em nuvem, onde foi criada uma interface de exibição e depuração destes dados, possibilitando o monitoramento em tempo real. Este projeto propõe uma solução econômica e acessível para fins educacionais e de pesquisa, visando contribuir com o estudo e o entendimento do sistema de energia elétrica em ambientes residências ou industriais.

Palavras-chave—Energia elétrica. Monitoramento. Qualidade de energia. Sensoriamento. Plataforma em nuvem.

I. INTRODUÇÃO

Para garantir a eficiência, qualidade e segurança de uma carga conectada em um sistema de rede elétrica é necessário que seja assegurado que os parâmetros de funcionamento estejam alinhados com as condições nominais de operação. À medida que as instalações e cargas se tornam mais complexas, a demanda por tecnologias que sejam capazes de realizar a análise desse funcionamento cresce, tornando muitas vezes o monitoramento da dinâmica das cargas no sistema, um fator indispensável. Além disso, como um sistema de energia elétrica é enxergado como um circuito fechado, onde cada elemento influencia diretamente na resposta deste circuito, buscar diagnósticos das interferências desses elementos neste sistema complexo é também buscar a melhor eficiência do todo, justificando, por exemplo, a criação do Programa de Eficiência Energética (PEE).

Porém nota-se que, muitos dos sistemas que realizam esse tipo de aferição apresentam altos custos iniciais, ou são destinados a aplicações específicas, limitando seu acesso para instalações mais simples, já que são essas que ocupam a maior parte das instalações do país. Com a popularização das linguagens de programação, avanços tecnológicos e o crescente interesse mundial na integração de plataformas físicas e em

nuvem, surge a necessidade de um equipamento de monitoramento que possa fornecer ao usuário o acesso tanto físico quanto remoto. Aliado a isso, o alcance a diferentes tipos de tecnologias, graças a globalização do comércio, é possível ter acesso a dispositivos que nos permitam o desenvolvimento de projetos que possam integrar todos esses conceitos.

II. REVISÃO BIBLIOGRÁFICA

A facilidade do acesso a tecnologias embarcadas e de baixo custo, como por exemplo microcontroladores e circuitos eletrônicos integrados impulsionam o desenvolvimento de sistema inteligentes [1].

A. Monitoramento de Energia Elétrica

O monitoramento de energia elétrica é uma atividade de grande importância para se analisar em uma rede elétrica. Por meio de análises desse monitoramento é possível inferir se o sistema em questão está trabalhando com seus parâmetros nominais e se as cargas conectadas estão sendo alimentadas corretamente. Além da questão intrínseca do funcionamento da rede, para o usuário destaca-se com grande importância a relação de consumo dessa energia paga; este monitoramento auxilia o consumidor na tomada de decisão e aumenta a economia de energia [2].

B. Microcontroladores

Microcontroladores são sistemas completos, como um pequeno computador, desenvolvido a partir de um circuito integrado, sendo possível a conexão com dispositivos de entrada e saída de dados [3].

Com o microcontrolador é possível a criação de um sistema embarcado, independente, capaz de realizar e processar tarefas. Esses dispositivos compactos possuem uma unidade de processamento central, memórias e demais periféricos que variam de acordo com o modelo da placa.

Dentre as placas mais utilizadas no mercado, destaca-se a ESP32, que é uma série de microcontroladores fabricados pela empresa Manufactor Espressif Systems, se destacando pelo seu baixo custo de aquisição e a disponibilidade de conexão sem fio, além de outras interfaces para periféricos interessantes. Na Figura 1, abaixo é mostrada a versão ESPWROOM32, utilizada neste projeto.

A integração das placas tipos ESP32 com a lógica de programação ambientada em sistemas tipos C/C++, juntamente com seu custo-benefício, são características que o tornam uma boa escolha para projetos que utilizam múltiplos módulos de processamento de dados, justificando sua escolha neste projeto.



Fig. 1. Microcontrolador ESP-WROOM-32 utilizado no projeto. Fonte [6]

C. Protocolo MQTT em sistemas IoT

Para que se torne possível a integração de diferentes tipos de dispositivos e plataformas, é necessária a utilização de um canal de comunicação que seja compartilhado entre todos esses dispositivos. Desta forma, destaca-se o protocolo de comunicação MQTT ("Message Queue Telemetry Transport"), usado como padrão de troca de mensagens entre o publicador e o leitor na mesma rede.

O protocolo MQTT tem seu surgimento datado em 1999, pela empresa IBM, com o objetivo de ser um mensageiro de alta precisão para redes limitadas [7].

Amplamente utilizado em sistemas eletrônicos conectados em rede, conhecido como IoT (Internet of things), gerando a dispositivos interfaces capazes de realizar troca de dados, tomada de decisões e impactando na interação do usuário com o equipamento [5]. Atualmente é o protocolo mais utilizado nestas aplicações, por ser confiável e de simples codificação, sustentando o uso dele neste trabalho.

D. Arduino

O Arduino é uma plataforma de código aberto, amplamente utilizada no mundo, para criação de projetos integrados com módulos eletrônicos, conhecido por sua fácil programação e utilização, até mesmo para aqueles com pouca familiaridade com o recurso [8].

Além do hardware produzido, o ambiente de programação Arduino é baseado em lógica de programação C/C++. Os programas são feitos baseados em controle de pinos digitais e analógicos, atuando como entrada ou saída de dados, simplificando operações feitas a partir da leitura de sensores e operações, sendo encontrado em sistemas de instrumentos musicais, robôs, jogos e sistemas interativos [9].

Desta forma, o Arduino é considerado uma ferramenta estratégica para desenvolvimento de projetos que necessitem da interação entre sensores e atuadores e processamento de dados,

características essas que justificam sua utilização neste projeto.

E. Thingable!

A Thingable! é uma plataforma para uso IoT, permitindo a integração de projetos e possibilitando a criação de aplicações dinâmicas que facilitam a leitura de dados e tomadas de decisão. O monitoramento das variáveis de projeto é possível a partir da utilização do protocolo MQTT.

Esta plataforma ainda conta com várias opções de personalização e padronização de eventos que tornam a programação mais fácil. Ela pode ser acessada de forma remota a partir de dispositivos conectados na internet, celulares ou computadores, além de permitir a extração de dados a partir de relatórios gerados.

Por ser uma plataforma democrática [10], seu uso se torna atrativo e confiável, defendendo sua aplicação neste trabalho.

F. Tensão

Tensão elétrica (V) é a força necessária para deslocar um elétron de um ponto ao outro [11]. Este deslocamento é devido a diferença de potencial existente entre esses dois pontos, diferença essa que surge do desbalanceamento de concentração de cargas em um material, determinando também o sentido desse deslocamento. A tensão pode ser expressa para valores eficazes ou RMS (Root Mean Square) e de pico; um valor eficaz de tensão é referente ao valor CA que entrega a mesma potência a uma carga para valores CC

$$V_{ef} = \sqrt{\frac{1}{T} \int_0^T v^2 * dt} \quad (1)$$

Por meio da equação (1) observamos que o valor eficaz é a raiz quadrada da média quadrada do sinal CA, por isso sua definição como RMS. O valor de pico é o valor máximo do sinal da onda alcançado durante um ciclo. No geral, em setores de energia elétrica o mais comum é a representação das magnitudes em valores RMS, pois é um indicador mais preciso.

G. Corrente

A corrente elétrica é o deslocamento de cargas em um intervalo de tempo, medida em amperes (A). Essa característica exclusiva de uma carga [11], permite que energia seja transportada de um ponto ao outro. Em um fio condutor quando há nele uma diferença de potencial, é forçado que as cargas deste material se desloquem de um lado ao outro no mesmo sentido, este fluxo é necessariamente o transporte de energia. A corrente também pode ser expressa em termos de valor de pico e RMS.

H. Potência Elétrica

Pode-se definir a potência elétrica como a taxa de transferência de energia, sendo relacionada com a variáveis de tensão e corrente. Em um circuito elétrico é de fundamental importância o entendimento do quando de potência manipula uma carga, para que se torne possível o estudo da dinâmica

deste elemento com toda a rede.

A primeira definição pertinente da potência é a instantânea, representado por meio dos valores de pico e ângulos de tensão e corrente.

$$p(t) = v(t) * i(t) = V * I * \cos(\omega t + \theta_{tensão}) * \cos(\omega t + \theta_{corrente}) \quad (2)$$

Como observado pela equação (2), a potência instantânea é uma variável variante no tempo, o que torna sua medida mais complexa que o desejado, para satisfazer as necessidades de medir a potência entregue a uma carga, e feita a média dessa medida de potência instantânea ao longo do tempo, chamada de potência média (P, medida em Watts).

$$P = \frac{1}{T} \int_0^T p(t) * dt \quad (3)$$

Essa medida é importante para o estudo de um circuito, pois nos fornece o valor real de potência da carga que efetivamente gera trabalho. Note que para ângulos de tensão e corrente em fase, é característica de circuito resistivo, e para valores defasados desses ângulos circuitos reativos, a potência média só faz referência a cargas resistivas.

Por outro lado, a multiplicação de valores eficazes de tensão e corrente nos fornece o resultado da potência aparente (S, medida em VA).

$$S = V_{RMS} * I_{RMS}^* \quad (4)$$

Esta variável também chamada de potência complexa, e a potência total fornecida pela fonte, parte real e conjugado.

Se temos a potência real e a potência total, fica claro que existe também a variável que diz respeito a parte imaginária de um circuito, na Figura 2, podemos ver a representação fasorial das potências.

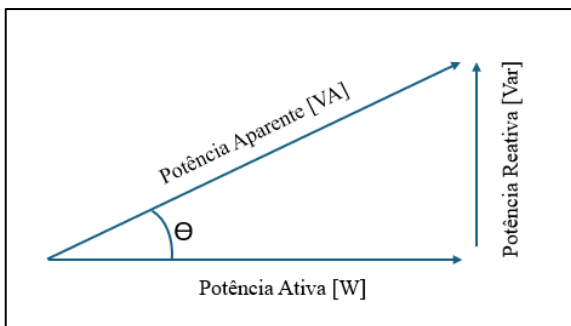


Fig.2.Triângulo das potências. Fonte: Autor

Dessa forma a potência reativa é dada por

$$Q = S * \sin(\theta) \quad (5)$$

I. Fator de Potência

Para determinar com qual eficiência a energia fornecida pela fonte é convertida em trabalho útil, é calculado a fator de potência desse sistema

$$fp = \frac{P}{S} = \cos(\theta) \quad (6)$$

Este valor adimensional indica também a eficiência do sistema. O valor ideal de fator de potência é 1, quando a potência total so sistema é somente ativa, casos de sistemas com cargas puramente resistivas. Com a inserção de elementos reativos no sistema, esse fator cai, indicando decaimento no desempenho do sistema.

III. METODOLOGIA

Para garantir a correta implementação do sistema, e a validação dos dados lidos e processados pelos sensores e algoritmo, o desenvolvimento do sistema proposto seguiu etapas metodológicas para sua especificação. Tais etapas são referentes à caracterização dos dispositivos e a final validação do sistema.

A. Materiais e Equipamentos

Foram selecionados sensores de tensão e corrente que se adequassem corretamente ao propósito do sistema a ser implementado, para isso foram levados em consideração parâmetros como precisão, compatibilidade com o microcontrolador selecionado e as faixas de leituras que poderiam ser realizadas. Desta forma, foram definidos para este projeto os sensores ACS712 [Fig. 3] e o ZMPT101B [Fig. 4], além do microcontrolador escolhido ESPWROOM32 [Fig. 1].



Fig. 3 Sensor de corrente ACS712. Fonte [12]

O sensor ACS712 funciona a partir do efeito hall, a corrente a ser medida passa dentro do dispositivo, onde internamente há uma corrente de excitação que gera um campo magnético, criando uma diferença de potencial entre os terminais do chip, proporcional à corrente que passa pelo condutor conectado.

Como não há contato físico entre o sensor e carga, uma vez que a medição é feita a partir do campo magnético, o ACS712 promove uma isolamento galvânica, importante para a proteção do

sistema.

Dessa forma, o valor analógico é lido e convertido para o valor da tensão, possibilitando o cálculo da corrente. Este sensor possui várias vantagens, como isolamento, medição bidirecional de corrente e uma largura de banda de até 80kHz, entretanto possui valores de offset que necessitam de calibração além de ser bem sensível a ruídos, principalmente quando o sistema está em aberto.

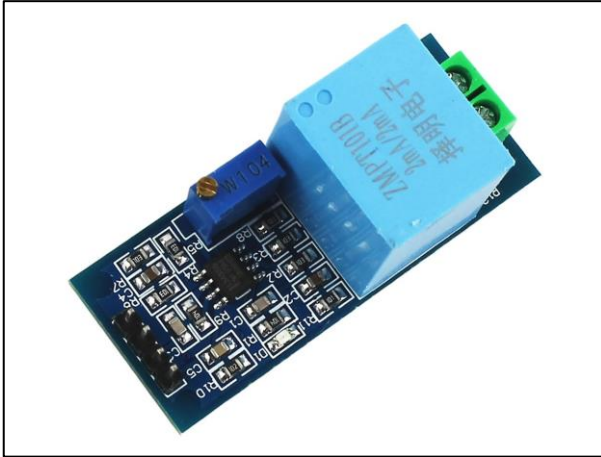


Fig. 4 Sensor de tensão ZMPT101B. Fonte [13].

O sensor de tensão ZMPT101B é um medidor de tensão elétrica que funciona a partir do funcionamento do TP (transformador de potencial) do seu circuito. Com valores de entrada que suportam até 250 V RMS, o TP abaixa a tensão de entrada para um valor seguro. O transformador funciona a partir do princípio de indução magnética, implicando um isolamento galvânico entre os lados de baixa e de alta tensão, promovendo também proteção ao sistema.

A saída deste sensor passa por um amplificador operacional que ajusta o valor para uma faixa adequada do ADC. Assim como o sensor de corrente, o ZMPT101B também possui offset, sendo necessária sua calibração para maior precisão de leitura.

A conexão dos sensores ao microcontrolador foi feita a partir da ligação das entradas analógicas da ESP às saídas de dados OUT. Os outros pinos referentes a entrada de alimentação e aterramentos dos sensores foram conectados aos pinos 3,3V e GND do microcontrolador, respectivamente.

A alimentação da ESP pode ser feita a partir de uma fonte de tensão contínua de 5 V nos pinos de alimentação da placa, ou a partir da conexão tipo-c à um dispositivo que aceite tal entrada. Na ligação deste circuito foram utilizados um protoboard e jumpers para realizar essas conexões.

Como esperado, a ligação dos sensores seguiu a lógica de ligação padrão de dispositivos de medição em um circuito, onde o medidor de corrente é ligado em série com a carga e o sensor de tensão ligado em paralelo, respeitando as indicações de sentido de fase e neutro de cada sensor. A Figura 5 apresenta o fluxograma do sistema.

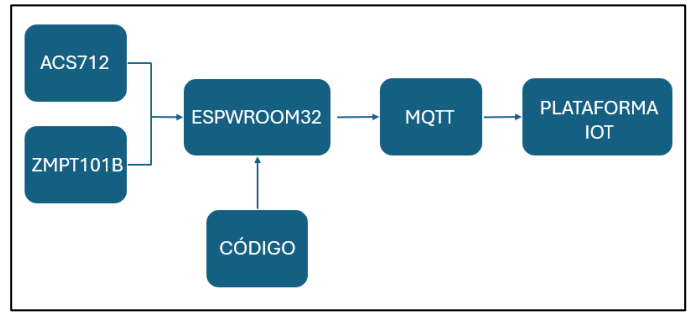


Fig.5 Fluxograma de conexão do modelo. Fonte: Autor.

O desenvolvimento do software para criação desta aplicação foi feito em Arduino, baseado nas seguintes etapas:

- Inclusão de bibliotecas para comunicação em rede;
- Definição dos pinos de entrada;
- Configuração do sistema e resolução ADC;
- Calibração dos sensores;
- Configuração de suavização e amostragem;
- Declaração das variáveis de medição e estado;
- Configuração das credenciais de rede;
- Configuração do broker MQTT;
- Cálculos dos valores RMS e potências;
- Publicação dos resultados.

B. Funcionamento do programa

Inicialmente é feita a aquisição e acúmulo das amostras dos sinais a serem processados, fazendo a leitura dos valores “brutos” e a conversão ADC-Tensão, levando em consideração a resolução de 12 bits da ESP e o valor de referência de 3,3 V, além disso, é feita a calibração dos sensores. Para o acúmulo dos valores RMS é feita a soma dos quadrados de cada amostra de corrente e tensão, e para a potência ativa faz-se a soma de valores instantâneos. Por fim, para minimizar efeitos de *aliasing*, é garantido 500 Hz de amostragem, maior que duas vezes o valor da frequência da rede; processo mostrado na Figura 6.

```

for (int i = 0; i < NUM_AMOSTRAS; i++) {
  int leitura_corrente_adc = analogRead(PINO_CORRENTE);
  int leitura_tensao_adc = analogRead(PINO_TENSAO);

  float voltagem_corrente = ((float)leitura_corrente_adc / ADC_RESOLUCAO) * VREF;
  float voltagem_tensao = ((float)leitura_tensao_adc / ADC_RESOLUCAO) * VREF;

  voltagem_corrente -= OFFSET_ACS;

  float corrente = -voltagem_corrente / SENSIBILIDADE_ACS;
  float tensao = (voltagem_tensao - VREF / 2.0) * FATOR_CALIBRACAO_TENSAO;

  soma_corrente_quadrado += corrente * corrente;
  soma_tensao_quadrado += tensao * tensao;
  soma_potencia_inst += corrente * tensao;

  delayMicroseconds((PERIODO_AMOSTRAGEM_MS * 1000) / NUM_AMOSTRAS);
}
  
```

Fig.6 Aquisição e acúmulo de amostras. Fonte: Autor.

Na sequência é feito os cálculos dos valores de tensão e potência, como exibidos nas equações (1), (4) e (3), processado pelo código conforme mostrado na Figura 7.

```
float corrente_rms_novo = sqrt(soma_corrente_quadrado / NUM_AMOSTRAS);
float tensao_rms_novo = sqrt(soma_tensao_quadrado / NUM_AMOSTRAS);
float potencia_aparente_novo = tensao_rms_novo * corrente_rms_novo;
float potencia_ativa_novo = soma_potencia_inst / NUM_AMOSTRAS;
```

Fig.7 Cálculo dos valores brutos. Fonte: Autor.

Para minimizar a interferência de valores advindos de ruídos nas leituras, foi feita uma suavização exponencial, filtro IIR de primeira ordem, estabelecendo uma constante de tempo discreta para um intervalo de 1 s, e constante alpha de 0,2, apresentado na Figura 8.

```
corrente_rms = (1 - ALPHA) * corrente_rms + ALPHA * corrente_rms_novo;
tensao_rms = (1 - ALPHA) * tensao_rms + ALPHA * tensao_rms_novo;
potencia_aparente = (1 - ALPHA) * potencia_aparente + ALPHA * potencia_aparente_novo;
potencia_ativa = (1 - ALPHA) * potencia_ativa + ALPHA * potencia_ativa_novo;
```

Fig.8 Suavização de valores. Fonte: Autor.

O fator de potência é calculado conforme demonstrado na equação (6), com os valores de potência ativa e aparente, implementado no código mostrado na Figura 9.

```
if (potencia_aparente > 0.1) {
    fator_potencia = potencia_ativa / potencia_aparente;
} else {
    fator_potencia = 0;
}
```

Fig.9 Cálculo do fator de potência. Fonte: Autor.

A energia consumida é atualizada conforme o tempo atual de operação, acumulando a cada iteração, desde o início da execução do *loop()* resultando em valor final em unidade Wh, exposto na Figura 10.

```
unsigned long tempo_atual = millis();
float delta_horas = (tempo_atual - tempo_anterior) / 3600000.0;
energia_consumida += potencia_ativa * delta_horas;
tempo_anterior = tempo_atual;
```

Fig.10 Cálculo de energia consumida. Fonte: Autor.

Por fim, todos os valores são publicados, conforme protocolo MQTT, a partir da publicação do cliente enviando um *payload* ao tópico do evento, como pode ser visto na Figura 11.

```
if (now - lastMeasure > 2000) {
    lastMeasure = now;
    sprintf(payload,
        "{\ntensao_rms\":%.2f,\n\"corrente_rms\":%.2f,\n\"potencia_aparente\":%.2f,\n\"potencia_ativa\":%.2f,\n\"energia_consumida\":%.2f,\n\"fator_potencia\":%.2f},\n",
        tensao_rms, corrente_rms,
        potencia_aparente, potencia_ativa,
        energia_consumida, fator_potencia);
    client.publish("Thingable/delufv/JAP_ESP32/JAP/evt", payload);
    Serial.println("[PUB] Topic: Thingable/... Msg: " + String(payload));
}
```

Fig.11 Publicação dos resultados via MQTT. Fonte: Autor.

Para a publicação dos dados obtidos pelo processamento realizado no programa desenvolvido, foi configurada uma rede de comunicação MQTT com a plataforma IoT escolhida Thingable!, para isso foram carregadas as credenciais de acesso

ao broker, como endereço, porta, usuário e senha. A rotina de leitura e publicação de resultados ocorre a cada 5 s, além da verificação em loop do estabelecimento da conexão da rede.

C. Configuração da Thingable!

Uma vez que o microcontrolador está apto para realizar as leituras e processamentos dos dados, assim como assumir a conexão com a rede, foi configurada a plataforma IoT para que ela pudesse reconhecer e mostrar os dados recebidos. Na Figura 12 é possível ver a tela inicial da plataforma, onde são feitas as principais configurações.

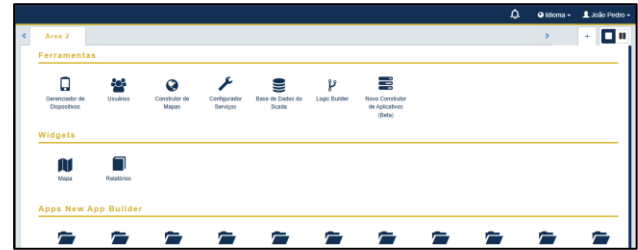


Fig.12 Página inicial Thingable!. Fonte: Autor.

Para o início da configuração é necessário a criação de um “Tipo”, que seria a definição da classe do seu dispositivo e o “Schema”, que seria o código em formato json, que contempla as variáveis recebidas do microcontrolador. Logo que criado o “Tipo” é necessário a configuração do “Dispositivo”, representando efetivamente o agrupamento de variáveis que deseja monitorar. Nas Figuras 13 e 14, são mostradas as telas de configuração dessas definições, na plataforma.



Fig.13 Configuração do “Tipo”. Fonte: Autor.

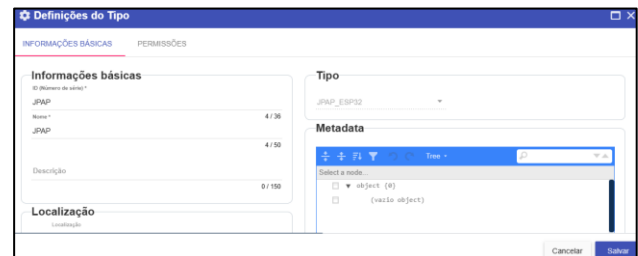


Fig.14 Configuração do “Dispositivo”. Fonte: Autor.

A criação do “Tipo” e “Dispositivo” é fundamental para caracterizar seu conjunto de variáveis lidas e criar as

credenciais que estabelecem o caminho de comunicação entre o objeto físico e a plataforma.

Finalizada essa primeira etapa, a Thingable! gera automaticamente, conforme estabelecido no seu “Schema”, um banco de dados com as variáveis instanciadas. Este banco de dados é importante pois define o código de cada uma dessas variáveis, que serão utilizadas para a criação do dashboard do aplicativo, apresentado na Figura 15.

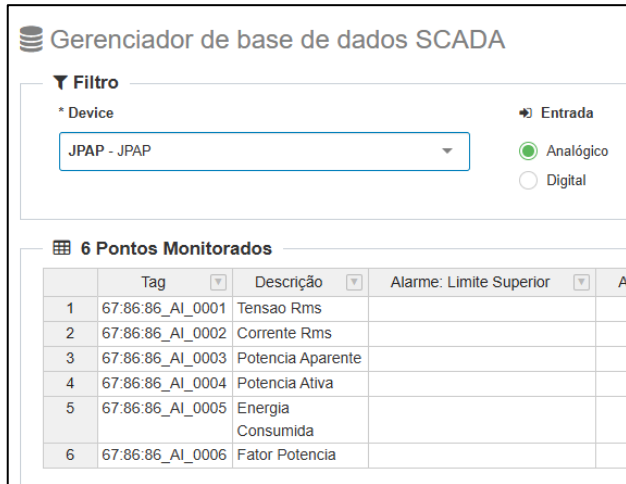


Fig.15 Gerenciador do banco de dados SCADA. Fonte: Autor.

A criação da janela de monitoramento dessas variáveis é feita a partir da função de criação de aplicativos da plataforma, onde é possível criar gráficos contínuos, medidores e tabelas para a apresentação dos dados lidos, chamado “Widgets”. Cada “Widget” leva em consideração a configuração da série registrada do dispositivo, ou seja, os códigos de cada variável do banco de dados do sistema.

IV. RESULTADOS E DISCUSSÃO

Para testar o funcionamento do sistema, assim como sua precisão, foram realizados testes com diferentes cargas conectadas. Dessa forma, foi testado o processamento do programa escrito e inserido na ESP, certificadas as conexões dos sensores com as cargas e com o microcontrolador e verificado se a conexão entre a plataforma Thingable! e o sistema físico ocorria sem interrupções. Nas Figuras subsequentes são mostradas as leituras a partir desses testes.

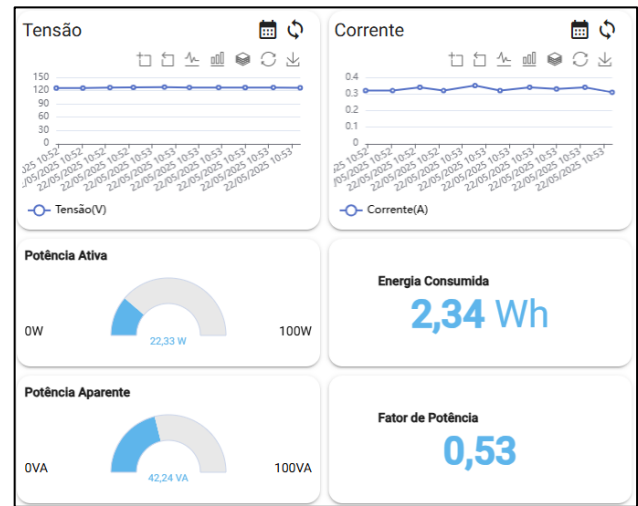


Fig.16 Interface de leitura criada na plataforma; sistema ligado a uma carga tipo refletor branco. Fonte: Autor.

O aplicativo criado é composto de dois gráficos que apresentam os valores de tensão e corrente no tempo, dois mostradores que apresentam as leituras de potência ativa e aparente seguido por outros dois mostradores que indicam os valores de energia consumida e fator de potência. Na Figura 16, são mostradas as leituras feitas para um refletor genérico. Apesar de indicar na caixa do produto potência de 100 W, foi verificado que este valor não bate com a leitura feita, entretanto, essa discrepância pode ser atribuída à indicação errônea do fabricante, comuns em produtos de baixo custo importados. Além disso, é também indicado na embalagem que o refletor trabalha com fator de potência superior a 0,6, indicando também certa diferença com o valor lido.

Na Figura 17, é mostrado teste realizado utilizando uma lâmpada halógena.

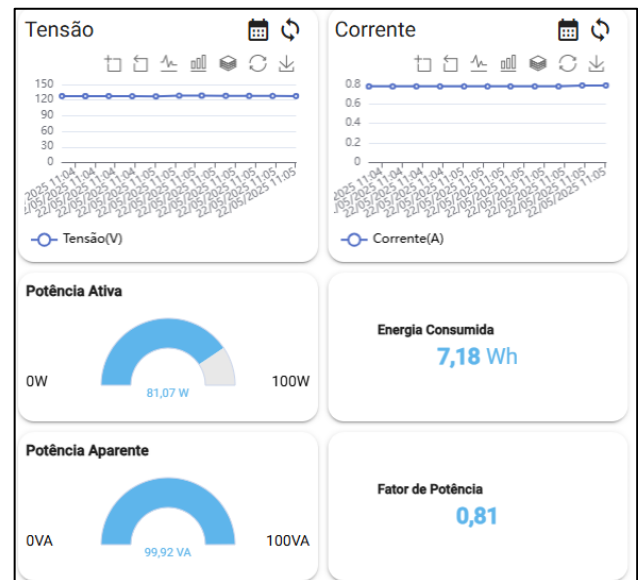


Fig.17 Interface com leitura realizada na conexão de uma lâmpada halógena de 100 W, durante 5 minutos. Fonte: Autor.

Como esta lâmpada pode ser considerada praticamente um elemento resistivo, o fator de potência esperado é próximo do unitário, como mostrado na Figura 17. Esta lâmpada é de 127

V, e em sua embalagem indicada potência de 100 W, como confirmado pelo programa.

Na sequência foi realizado um teste em uma geladeira da marca Electrolux, modelo RE3, ajustado para funcionar no nível de temperatura Médio. As Figuras abaixo apresentam diferentes momentos desse teste.



Fig.18 Geladeira ligada no nível de temperatura ajustado para máximo, após 1:35 h de funcionamento, com compressor ligado. Fonte: Autor.

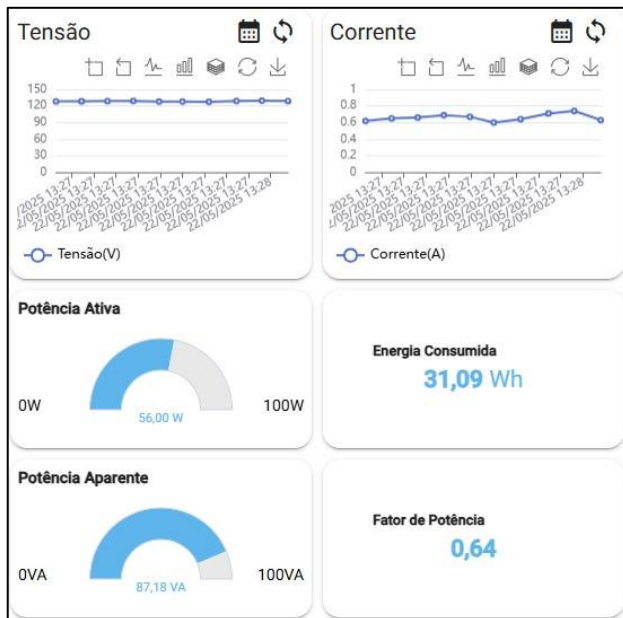


Fig.19 Geladeira ligada no nível de temperatura médio, após 1:48 h de funcionamento, com compressor ligado. Fonte: Autor.



Fig.20 Geladeira ligada no nível de temperatura médio, após 2:20 h de funcionamento, com o compressor desligado. Fonte: Autor.



Fig.21 Geladeira ligada no nível de temperatura médio, após 6 h de funcionamento. Fonte: Autor.

As Figura 18 a 20 mostram os níveis lidos pelo sistema para diferentes estágios de funcionamento da geladeira, carga em questão. É possível observar que os valores são realistas, levando em consideração por exemplo que a fabricante garante potência ativa máxima de 75 W. Pode-se perceber as diferenças nos valores quando o compressor está ligado ou desligado, como esses motores possuem comportamento indutivo, os valores lido de fator de potência também estão como esperado para esta carga.

Na Figura 21 é possível observar o comportamento das variáveis calculadas durante o período de 6 h de funcionamento ininterruptos. Os picos que podem ser observados nos gráficos são os intervalos onde o compressor liga, e como esperado, nesses intervalos de tempo o motor consumirá mais corrente da rede, influenciando nos valores de potência e o fator de aproveitamento ativo da geladeira. O gráfico da energia consumida, também observado na Figura 21, mostra o consumo que esta geladeira teve neste período de teste, próximo ao valor de 90 Wh.

V. CONCLUSÃO

O programa desenvolvido neste projeto mostrou-se eficaz na realização de leitura e processamento dos dados. Conforme

foram sendo realizadas leituras de teste, tanto para os sensores de corrente como para o sensor de tensão foram encontrados os valores de offset de calibração deles, a partir da comparação de suas medições com a valores lidos em multímetros presentes em laboratório.

A plataforma Thingable! se mostrou muito eficaz para o monitoramento das variáveis elétricas nas cargas testadas, fazendo com que se torne fácil a visualização dos parâmetros calculados, assim como a identificação de possível problemas a partir dos gráficos gerados.

Com os testes realizados com as diferentes cargas, foi possível perceber que o sistema apresenta um alto índice de confiabilidade, mostrando boa precisão, mesmo sendo composto pelos sensores mais baratos da categoria, indicando que tratamento de dados realizado internamente pelo programa criando, assim como a suavização das leituras teve fundamental importância para compensar possíveis imprecisões causadas por ruídos.

Por fim, este trabalho comprovou que uma solução de monitoramento embarcado baseado em ESP32, sensores ACS712 e ZMPT101B juntamente com a plataforma Thingable! é capaz de fornecer dados confiáveis e em tempo real sobre variáveis elétricas em cargas variadas, mesmo sendo um projeto de baixo custo. Apesar dos excelentes resultados, esse sistema pode ser melhorado trocando por sensores de maior precisão, com um custo mais elevado. Ademais este projeto pode ser ampliado, adicionando diversas outras melhorias no futuro, como a implementação de um relé para controle de conexão das cargas na rede, podendo ser por exemplo baseado em lógica de sobrecorrente ou limite do consumo, adicionar um sistema de classificar de eventos elétricos, com mensagens de alerta enviados pelo aplicativo ao usuário, entre outras diversas possibilidades de upgrade. Com essas evoluções espera-se ampliar ainda mais a robustez, a autonomia e o potencial de aplicação do sistema em ambientes industriais e de pesquisa.

REFERÊNCIAS

- [1] BALL, Stuart. *Embedded Microprocessor Systems: Real World Design: Embedded Technology Series*. 3. ed. USA: Newnes, 2002. 378 p.
- [2] PASCALICCHIO, Agostinho. *PERPECTIVA ECONÔMICA E MODELO DE NEGÓCIO DA TECNOLOGIA DE TELECOMUNICAÇÃO NAS REDES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA NO BRASIL*. Pós-graduação em Energia da Universidade de São Paulo, [S. l.], p. 1-147, 1 jan. 2011.
- [3] ARAÚJO, Warley *et al.* Visão geral sobre microcontroladores e prototipagem com arduino. *Revista Tecnologias em Projeção*, [S. l.], p. 1-11, 1 jan. 2019.
- [4] OLIVEIRA, Bruno. *DESENVOLVIMENTO DE UM PROTÓTIPO DE SISTEMA DE MONITORAMENTO DE ENERGIA ELÉTRICA VIA INTERNET*. Programa de pós-graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais, Belo Horizonte, Brasil, p. 1-14, 1 jan. 2018.
- [5] DA SILVA, Danyllo Valente *et al.* Uma tecnologia para apoiar a engenharia de requisitos de sistemas de software iot. In: 23rd Iberoamerican Conference on Software Engineering. 2020.

[6] DOIT ESP32 DEV KIT v1: high resolution pinout and specs. [S. l.], 17 fev. 2021. Disponível em: <https://mischianti.org/doit-esp32-dev-kit-v1-high-resolution-pinout-and-specs/>. Acesso em: 30 abr. 2025.

[7] SONI, Dipa; MAKWANA, Ashwin. A survey on mqtt: a protocol of internet of things (iot). In: *International conference on telecommunication, power analysis and computing techniques (ICTPACT-2017)*. 2017. p. 173-177.

[8] BANZI, Massimo. *Getting Started With Arduino*. [S. l.: s. n.], 2011.

[9] MARGOLIS, Michael. *Arduino CookBook*. [S. l.: s. n.], 2011.

[10] Bem vindo a wiki da thingable!. [S. l.], 30 abr. 2024. Disponível em: <https://docs.thingable.com/bem-vindo-a-wiki-da-thingable>. Acesso em: 30 abr. 2025.

[11] SADIKU, Mattew. *Fundamentos de Circuitos Elétricos*. 5. ed. rev. [S. l.: s. n.], 2013.

[12] SENSOR de Corrente ACS712 5A AC / DC com Efeito Hall. [S. l.], 5 maio 2025. Disponível em: <https://www.usinainfo.com.br/sensor-de-corrente-arduino/sensor-de-corrente-ac-712-5a-ac-dc-com-efeito-hall-2550.html>. Acesso em: 5 maio 2025.

[13] SENSOR de Tensão AC 0-250V Zmpt101b / Voltímetro para Arduino. [S. l.], 5 maio 2025. Disponível em: <https://www.usinainfo.com.br/sensor-de-tensao-arduino/sensor-de-tensao-ac-0-250v-zmpt101b-voltmetro-para-arduino-5658.html>. Acesso em: 5 maio 2025.

JOÃO PEDRO ALMEIDA PINTO

**DESENVOLVIMENTO DE UM SISTEMA EMBARCADO PARA
MONITORAMENTO DE ENERGIA ELÉTRICA**

Monografia apresentada ao Departamento de Engenharia Elétrica do Centro de Ciências Exatas e Tecnológicas da Universidade Federal de Viçosa, para a obtenção dos créditos da disciplina ELT 402 – Projeto de Engenharia II – e cumprimento do requisito parcial para obtenção do grau de Bacharel em Engenharia Elétrica.

Aprovada em 27 de junho de 2025.

COMISSÃO EXAMINADORA



Prof. Dr. André Gomes Torres - Orientador
Universidade Federal de Viçosa



Prof. Dr. Mauro de Oliveira Prates - Membro
Universidade Federal de Viçosa



Prof. Dr. Denilson Eduardo Rodrigues - Membro
Universidade Federal de Viçosa