

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO

UM ESTUDO SOBRE MEDIDORES INTELIGENTES DE ENERGIA

Rafael Givanildo da Silva

São Carlos
2021

RAFAEL GIVANILDO DA SILVA

UM ESTUDO SOBRE MEDIDORES INTELIGENTES DE ENERGIA

Trabalho de conclusão de curso apresentado ao Departamento de Computação da Universidade Federal de São Carlos, para obtenção do título de bacharel em Engenharia de Computação.

Orientador: Prof. Dr. Emerson Carlos Pedrino.

São Carlos

2021

Um estudo sobre medidores inteligentes de energia.

Rafael Givanildo da Silva

Trabalho de conclusão de curso
apresentado ao Departamento de
Computação da Universidade Federal de São
Carlos, para obtenção do título de bacharel
em Engenharia de Computação.

Aprovado em: 13 / 01 / 2021.

BANCA EXAMINADORA

Orientador

Prof. Dr. Emerson Carlos Pedrino
Universidade Federal de São Carlos

Membro da banca (1)

Prof. Dr. Fredy João Valente
Universidade Federal de São Carlos

Membro da banca (2)

Prof. Dr. Valentin Obac Roda
Universidade Federal do Rio Grande do Norte

Dedico aos meus pais, pela dedicação, amor e paciência. Dedico aos meus amigos de alojamento pela paciência, apoio e perseverança por esses anos.

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus, pela minha vida e por me ajudar durante todos estes anos. Agradeço por todas as graças e oportunidades que recebi durante estes anos.

Agradeço aos meus pais, por todo apoio e ajuda, por todo o incentivo que tive e tenho durante essa jornada.

Ao Professor Emerson agradeço por ter sido meu orientador desde o início. Obrigado por todo apoio, dedicação e amizade. Agradeço ao Denis por todas as dicas e orientação.

Por fim, agradeço a todos os meus amigos, em especial ao Marcos, ao Edgar e ao Romão.

“Nossa maior fraqueza é a desistência. O caminho mais certo para o sucesso é sempre tentar apenas uma vez mais”

THOMAS ALVA EDISON [26]

RESUMO

O setor elétrico mundial está passando por uma grande revolução graças ao nascimento e implementação das chamadas *Smart Grids* ou Redes Inteligentes de Energia. Essa revolução é centrada no uso massivo e ativo dos dados gerados por essas redes, utilizando-se de conceitos como, Internet das Coisas, *Big Data* e Inteligência Artificial. Um dos equipamentos chave para essa revolução são os medidores inteligentes de energia, que coletam e enviam os dados que tornam a rede “inteligente”.

Esse trabalho tem como objetivo apresentar o desenvolvimento de um protótipo de medidor de energia, baseado no conceito de internet das coisas, de baixo custo e *open source*.

Palavras-chave: Internet das Coisas. Redes inteligentes de energia. Medidores inteligentes de energia. Infraestrutura de medição avançada.

ABSTRACT

The world electricity sector is undergoing a major revolution thanks to the birth and implementation of so-called Smart Grids or Intelligent Energy Grids. This revolution is centered on the massive and active use of data generated by these networks, using concepts such as Internet of Things, Big Data and Artificial Intelligence. One of the key equipment for this revolution is the smart energy meters, which collect and send data that make the network “smart”.

This work aims to present the development of a prototype energy meter, based on the concept of internet of things, low cost and open source.

Keywords: *Internet of Things. Smart Grid. Smart Meter. Advanced metering infrastructure.*

Lista de ilustrações

Figura 1 - Ilustração de diversas fontes de energia que compõe a <i>Smart Grid</i>	15
Figura 2 - Imagem que resume as motivações da implantação das REIs pelo mundo.	16
Figura 3 - Ilustração da arquitetura de rede presente na AMI.	20
Figura 4 - Apresenta o funcionamento da tarifa branca durante os dias úteis (esquerda) e finais de semana/feriados. Além disso, é possível fazer uma comparação entre as tarifas existentes.	22
Figura 5 - Ilustração que ilustra o funcionamento do protocolo MQTT.	24
Figura 6 - Ilustração de uma comunicação SPI composta por diversos escravos.	25
Figura 7 - Ilustração do funcionamento da comunicação SPI.	26
Figura 8 - Diagrama interno do chip.	27
Figura 9 - Diagrama dos pinos do CI.	28
Figura 10 - Diagrama de funcionamento lógico 1.	29
Figura 11 - Diagrama de funcionamento lógico 2.	30
Figura 12 - Formato de um comando de escrita ou leitura.	31
Figura 13 - Diagrama de conexão, monofásica e conexão direta com a rede de energia.	32
Figura 14 - Diagrama de conexão, monofásico com isolamento.	32
Figura 15 - Intel Edison e a placa Breakout Arduino.	33
Figura 16 - Transformador de corrente YHDC SCT-013-000.	34
Figura 17 - Sensor instalado.	35
Figura 18 - Transformador MAXXI 12 + 12 V 300mA utilizado.	35
Figura 19 - Display LCD RGB I2C e Base Shield.	36
Figura 20 - Arquitetura proposta pela Cirrus Logic no documento de design.	37
Figura 21 - Diagrama de blocos - Medidor Inteligente - ST.	37
Figura 22 - Diagrama de conexão escolhido.	38
Figura 23 - Diagrama do circuito impresso (protótipo).	39
Figura 24 - Esquemático final.	40
Figura 25 - Placa finalizada.	40
Figura 26 - Diagrama de classes.	43
Figura 27 - Fluxogramas explicando como realizar a medição de uma grandeza elétrica e como realizar o cálculo da tarifa.	44
Figura 28 - Página inicial da dashboard.	45

Lista de Tabelas

Tabela 1 - Tabela de componentes e seus preços aproximados.	46
Tabela 2 - Tabela comparativa entre as medições realizadas pelo protótipo e os multímetros.....	47

SUMÁRIO

1 INTRODUÇÃO	12
1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO	12
1.2 OBJETIVOS.....	12
1.3 ESTRUTURA DO TRABALHO	13
2 REVISÃO BIBLIOGRÁFICA	14
2.1 REDES INTELIGENTES DE ENERGIA - <i>SMART GRID</i>	14
2.2 MEDIDOR INTELIGENTE DE ENERGIA – <i>SMART METER</i>	17
2.3 INFRAESTRUTURA AVANÇADA DE MEDIÇÃO – <i>ADVANCED METERING INFRASTRUCTURES - AMI</i>	19
2.4 TARIFA BRANCA	21
3 PROTÓTIPO DESENVOLVIDO	23
3.1 REVISÃO TEÓRICA	23
3.1.1 <i>UPM</i>	23
3.1.2 <i>LIBMRAA</i>	23
3.1.3 <i>MQTT</i>	24
3.1.4 <i>SPI</i>	25
3.2 HARDWARE	27
3.2.1 <i>CS5463</i>	27
3.2.2 <i>Intel Edison</i>	33
3.2.3 <i>Sensor de corrente</i>	34
3.2.4 <i>Sensor de tensão</i>	35
3.2.5 <i>LCD e Base Shield</i>	36
3.3 ARQUITETURA PROPOSTA	36
3.3.1 <i>Hardware</i>	36
3.3.2 <i>Software</i>	41
4 RESULTADOS E DISCUSSÃO	46
5 CONCLUSÃO	49
REFERÊNCIAS	50
APÊNDICE A – PRODUÇÃO TÉCNICO-CIENTÍFICA	53

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO

Desde a invenção das redes elétricas, não houve mudanças significativas na tecnologia utilizada para geração, transmissão e distribuição de energia elétrica, onde as tecnologias usadas remontam ao final do século XIX [1].

Atualmente, necessita-se fortemente da energia elétrica, sendo ela base para praticamente todos os sistemas e atividades existentes (direta ou indiretamente). A falta de eletricidade gera diversos inconvenientes para a sociedade, governos e organizações afins. Essa questão gera prejuízos de ordem econômica, de mobilidade e de segurança.

Visando trazer as redes de energia para o século XXI e responder aos desafios existentes atualmente, nasce em 2005, o conceito de *Smart Grid*/Redes inteligentes de Energia/REI. Esse termo foi cunhado por S. Massoud Amin e Bruce F. Wollenberg.

Segundo Amin & Wollenberg [2], o conceito de *Smart Grid* se baseia na racionalização dos sistemas de geração, transmissão e distribuição utilizando a eletrônica e a ciência da computação em busca de uma rede dita inteligente. Com a aplicação do conceito de *Smart Grid* é possível uma melhor gestão da rede, a introdução de novas formas de geração de energia e a descentralização da geração. Além disso, o consumidor poderá se beneficiar de uma maior quantidade de informações, que poderão servir para diminuir o seu consumo.

Uma das tecnologias chaves para a implementação desse conceito são os medidores inteligentes de energia, que são o foco deste trabalho. Estes equipamentos são responsáveis por coletar e enviar os dados da rede, além de servir de base para diversos novos serviços e funcionalidades.

1.2 OBJETIVOS

Esse trabalho tem como objetivo apresentar o desenvolvimento de um protótipo de medidor inteligente de energia para uso residencial ou em pequenos comércios.

Os objetivos específicos do trabalho são:

- Apresentar uma pequena revisão bibliográfica a respeito dos seguintes conceitos:
 - Redes Inteligentes de Energia / *Smart Grid*;
 - Medidor Inteligente de Energia / *Smart Meter*;
 - Infraestrutura Avançada de Medição / *Advanced Metering Infrastructure / AMI*.
- Apresentar de forma detalhada como foi feita a implementação do protótipo de medidor de energia.

1.3 ESTRUTURA DO TRABALHO

A estrutura do trabalho está dividida da seguinte forma:

- Capítulo 1: apresenta a contextualização, as motivação e os objetivos do trabalho;
- Capítulo 2: apresenta uma revisão bibliográfica de conceitos importantes;
- Capítulo 3: apresenta detalhadamente o desenvolvimento do protótipo de medição;
- Capítulo 4: apresenta os resultados obtidos;
- Capítulo 5: apresenta as conclusões obtidas ao término do desenvolvimento.

2 REVISÃO BIBLIOGRÁFICA

2.1 REDES INTELIGENTES DE ENERGIA - *SMART GRID*

Antes de apresentar o conceito de Redes Elétricas Inteligentes (REI) é interessante apresentar como é o setor elétrico. O sistema basicamente é composto por grandes unidades geradoras de energia, distantes dos grandes centros de consumo fazendo com que as redes de transmissão sejam muito longas e caras. Essa rede é unidirecional e não é totalmente automatizada.

Segundo Aboboreira & Cruz [3], os sistemas de geração e transmissão no Brasil, apresentam algum grau de automação, supervisão e controle. Diferentemente desses sistemas, os sistemas de distribuição estão num estágio anterior de implementação, devido ao elevado número de clientes e a grande complexidade dos sistemas.

O setor de energia é o maior responsável global por emissões de gases de efeito estufa (29% das emissões) [4], devido ao uso em larga escala de fontes de energia não renováveis. Além da questão ambiental, o setor sofre com perdas (técnicas e não técnicas) durante a transmissão e distribuição e com os altos custos de operação, manutenção e expansão.

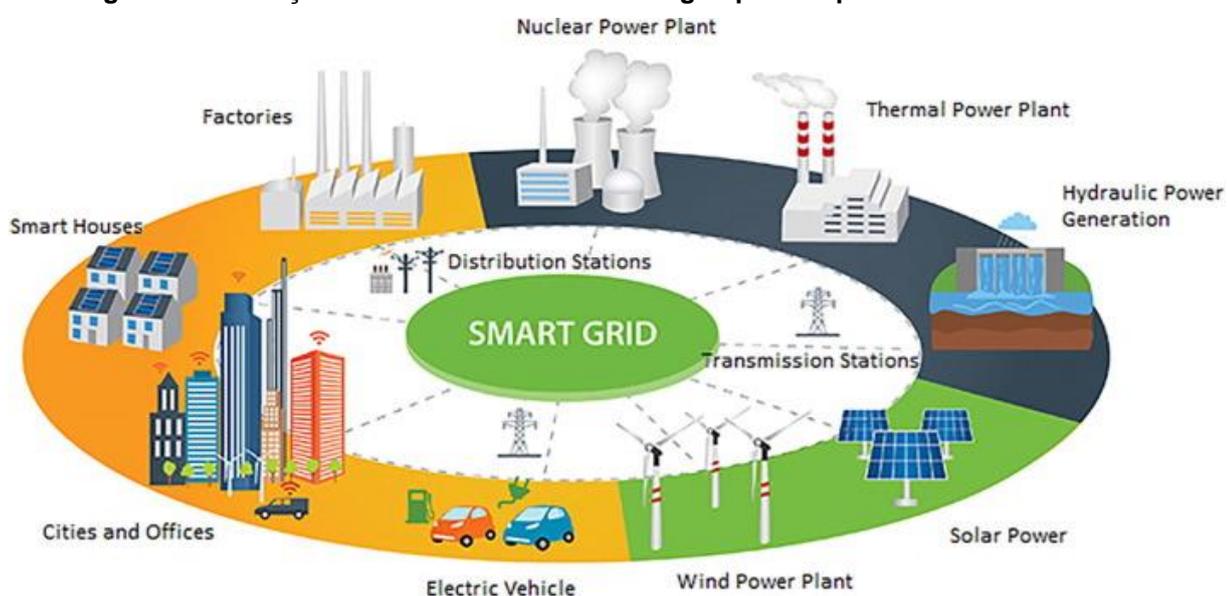
Outro ponto que vale ser pontuado é que vivemos num panorama em que é imprescindível para as sociedades contemporâneas a disponibilidade da energia elétrica, pois ela está intimamente ligada a maioria das atividades. Infelizmente, as redes clássicas não são resilientes e ágeis, fazendo com que problemas relacionados a geração (demanda em excesso) e a transmissão/distribuição (queda ou problema em equipamentos) sejam facilmente sentidos pelo consumidor final.

Amin & Wollenberg cunharam numa publicação da IEEE (*Institute of Electrical and Electronics Engineers*) o termo *Smart Grid*, ou em português Redes Elétricas Inteligentes. Na publicação os autores definem o termo como: “infraestrutura de rede elétrica em larga escala caracterizada por segurança, agilidade e resiliência/robustez que enfrenta novas ameaças e condições não previstas” [2]. O termo cunhado por eles, significou uma grande mudança de paradigma, algo que vai além da simples implementação de uma determinada tecnologia.

A NIST (*National Institute of Standards and Technology*) define Redes Eléctricas Inteligentes como: “Rede eléctrica que usa informações, de fluxo bidirecional, com tecnologias de comunicação segura e inteligência artificial de forma a integrar todo o espectro do sistema de energia, desde a geração até o consumidor final” [5].

Além das duas definições apresentadas existem diversas outras, que seguem o mesmo cerne, mais que acrescentam uma visão local ou uma necessidade específica do autor ou entidade. Abaixo, segue uma imagem que ilustra como se dá esse conceito de rede eléctrica, onde há uma integração de diversas fontes de energia e atores que compõem o conceito.

Figura 1 - Ilustração de diversas fontes de energia que compõe a *Smart Grid*.



Fonte: Eolas [27]

A implementação do conceito de *Smart Grid* implica tanto na modernização da infraestrutura de rede, envolvendo todas as questões tecnológicas inerentes, quanto na mudança de atitude e de processos dos agentes que compõem as redes de energia.

Segundo o Centro de Gestão e Estudos Estratégicos - CGEE [6], usando como base um documento da IEA de 2011, lista as seguintes características para as Redes Inteligentes. São elas:

- Auto recuperação da rede em caso de falha;
- Participação proativa dos consumidores;

- Tolerância a ataques externos;
- Foco na qualidade de energia;
- Capacidade para acomodar uma grande variedade de fontes de energia;
- Menor impacto ambiental;
- Atuação remota, visando resposta em demanda;
- Viabilizar e beneficiar a competição no mercado de energia.

Os países desenvolvidos foram os primeiros a iniciarem a transformação de suas redes elétricas convencionais em inteligentes, assumindo o papel de liderança e de disseminadores das tecnologias e experiências para os países em desenvolvimento.

Pelo mundo houve diversas motivações para justificar os investimentos em REI. A seguir, segue uma ilustração retirada de um estudo do BNDES [1], que resume os principais motivos das nações desenvolvidas investirem na implementação das REI.

Figura 2 - Imagem que resume as motivações da implantação das REIs pelo mundo.

EUA	Europa	Japão	China
<ul style="list-style-type: none"> • Agenda tecnológica para recuperação econômica • Infraestrutura obsoleta • Geração distribuída de energia • Confiabilidade, segurança e eficiência do sistema • Uso de veículos elétricos e híbridos 	<ul style="list-style-type: none"> • Integração de diversas fontes de energia renováveis • Infraestrutura envelhecida • Uso de veículos elétricos 	<ul style="list-style-type: none"> • Diversificação energética (acidentes nucleares) • Uso de veículos elétricos • Implantação de cidades inteligentes 	<ul style="list-style-type: none"> • Implantação de cidades inteligentes e protagonismo mundial em IoT • Eficiência energética • Diversificação energética (renováveis)

Fonte: BNDES [1]

Segundo a Aneel (Agencia Nacional de Energia Elétrica) os motivos para a implantação das REI no Brasil são: busca por eficiência comercial e energética, aumento da confiabilidade da rede, segurança operacional e de sistema, e sustentabilidade econômica e ambiental [1].

O Brasil, assim como uma grande parte dos países em desenvolvimento está em um estágio inicial na implementação do conceito de Redes Inteligentes de Energia. Segundo estudos da Abradee (Associação Brasileira de Distribuidores de Energia

Elétrica), serão necessários investimentos entre R\$ 46 bilhões a R\$ 91 bilhões até 2030 para conseguir implementar o conceito de REI no Brasil [7].

Alguns dos benefícios da REI são [6]:

- aumento na segurança das redes de energia, devido ao fato da rede ser bidirecional e distribuída;
- diminuição das perdas não técnicas, devido a inclusão de novos medidores e sensores, permitem uma maior detecção de fraudes;
- possibilidade do consumidor se tornar um micro gerador de energia, possibilitando uma diminuição na conta deste e melhora na resiliência da rede;
- automatização das redes de energia;
- maior transparência entre as distribuidoras e os consumidores;
- possibilidade de novos modelos de tarifação.

Alguns dos desafios para a implementação das Redes Inteligentes são [6]:

- Estabelecer um padrão de interoperabilidade e segurança entre os equipamentos;
- Desenvolver um modelo de mercado viável, devido ao alto custo de implementação;
- Promover a *cyber* segurança do sistema;
- Promover a privacidade dos dados dos clientes.
- Engajar os clientes para o uso consciente da energia.

2.2 MEDIDOR INTELIGENTE DE ENERGIA – *SMART METER*

Da mesma forma que as redes de energia convencionais são baseadas em tecnologias do final do século XIX, os medidores de energia também padecem desse mesmo cenário. Para apresentar o cenário atual dos medidores de energia e sua importância nas REI é interessante mostrar como se deu a evolução dos medidores de energia e o que nos levou ao conceito de Medidores Inteligentes de Energia.

O modelo de medidor mais consagrado é o eletromecânico, que se baseia no princípio da indução eletromagnética e foi inventado por Shallenberger em 1888 [8].

Esse tipo de medidor mede somente o consumo de energia ativa e para a aferição da tarifa é necessária a coleta manual dos dados – isso demonstra um fluxo unidirecional da informação.

Os medidores eletromecânicos apresentam problemas na precisão da medição em situações de variação na temperatura ambiente, na tensão de rede, na frequência ou na presença de harmônicos ou campos magnéticos externos. Além disso, podem ocorrer problemas devido ao desgaste de seus componentes internos – a vida útil desse tipo de medidor pode chegar a 25 anos [9].

Com o desenvolvimento dos sistemas digitais e com seu posterior barateamento, surgiram os medidores eletrônicos de energia. Esse tipo de medidor baseia-se no uso em conjunto de conversores A/D e microcontrolador ou microprocessador.

Em geral os medidores eletrônicos são mais precisos que os eletromecânicos. Esse tipo de medidor tem como desvantagens:

- vida útil ser entre 13 e 15 anos (menor que o eletromecânico);
- insegurança quanto ao seu funcionamento em amplas condições climáticas;
- custo maior que os medidores eletromecânicos [8].

Com o passar do tempo os medidores eletrônicos evoluíram para o que hoje são chamados de Medidores Inteligentes de Energia ou *Smart Meters*. A *Edison Electric Institute* (EEI) define os *Smart Meters* como: “dispositivos de medição eletrônica, usado por concessionárias de energia para a transmissão de informações, para a cobrança dos consumidores e para a operação dos sistemas elétricos” [10]. Esse tipo de medidor foi introduzido inicialmente para grandes consumidores, devido as necessidades especiais de tarifação. Com o barateamento da tecnologia e com os requisitos impostos pelas REI, esse tipo de medidor começou a ser utilizado pelos demais consumidores.

Os Medidores Inteligentes de Energia permitem as seguintes funcionalidades:

- telemetria e faturamento mais precisos;
- monitoramento de aspectos ligados a qualidade da rede;
- possibilitar a geração distribuída por pequenos consumidores;
- corte e religamento remoto do fornecimento de energia;

- detecção automática de falta de energia;
- detecção facilitada de fraudes no medidor;
- possibilidade de diversos tipos de tarifação (tarifa em tempo real, tarifas horo-sazonais, tarifa pré-paga e tarifa de picos críticos);
- integração com outros dispositivos – medidores de gás e/ou água e dispositivos domésticos inteligentes.

As desvantagens dos Medidores Inteligentes de Energia estão relacionadas com o custo e com a durabilidade do equipamento. A questão relacionada ao custo, é ponto especial nos *smart meters*, pois eles podem custar até oito vezes mais que os medidores eletromecânicos [1].

No Brasil há por volta de 70 milhões de medidores, onde 90% são eletromecânicos [7], tornando imprescindível a substituição desses medidores por *smart meters*.

2.3 INFRAESTRUTURA AVANÇADA DE MEDIÇÃO – *ADVANCED METERING INFRASTRUCTURES - AMI*

Segundo a consultoria Gartner [11], a AMI é definida como um conjunto de tecnologias compostas pelos medidores inteligentes de energia, sistemas de comunicação bidirecionais e um repositório/gerenciador de dados.

A AMI é base fundamental para a implementação do conceito de *Smart Grid*, pois é a camada que gerencia e alimenta de dados os demais sistemas existentes nas distribuidoras.

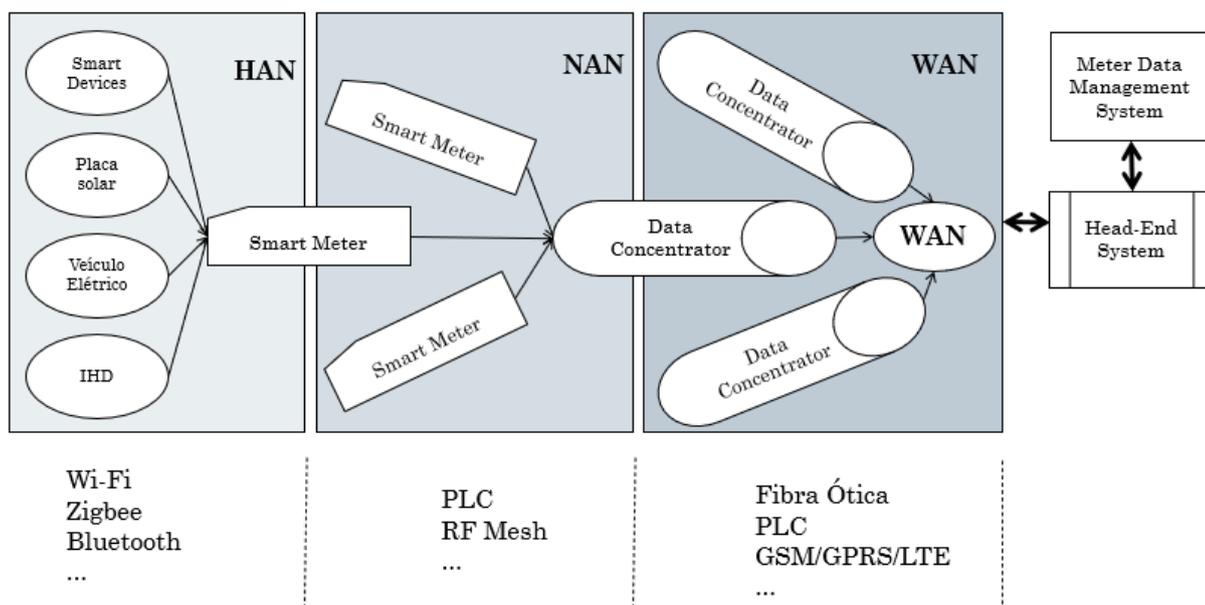
Segundo Mohassel, Fung, Mohammadi & Raahemifar [12], a AMI é composta pelos medidores inteligentes de energia, diferentes camadas de comunicação e o sistema de gerenciamento de dados de medição (MDMS).

Os *Smart Meters* têm como função coletar e enviar os dados para as distribuidoras. Além disso, esses medidores tem comunicação bidirecional, possibilitam novas funcionalidades, citadas na seção específica sobre medidores inteligentes.

Os medidores podem utilizar diferentes tecnologias de comunicação, como por exemplo: PLC, Zigbee, RF Mesh ou redes móveis, para estabelecer conexão com os concentradores - ou diretamente com a distribuidora dependendo do caso.

Olhando para as camadas de comunicação, a AMI é composta por três tipos de redes, são elas: *home area network* - HAN, *neighbourhood area network* - NAN e *wide area network* – WAN [13]. A imagem a seguir ilustra a arquitetura descrita.

Figura 3 - Ilustração da arquitetura de rede presente na AMI.



Baseado em: MARIMUTHU, Krishna Paramathma; DURAIRAJ, Devaraj; SRINIVASAN, Swathikumar Karthik [28]

A HAN é uma rede gerada pelos *smart meters* responsável por conectar o medidor inteligente a dispositivos inteligentes, casas inteligentes, sistemas de geração e armazenamentos locais, carros elétricos e IHD. Essa rede pode utilizar: *Wi-fi*, *Zigbee* e *Bluetooth*.

A NAN é a rede responsável por concentrar os dados dos medidores existentes na vizinhança e enviá-los para a distribuidora via WAN. Nessa etapa são utilizados concentradores, que concentram os dados dos *smarts meters*. A comunicação dos medidores com o concentrador pode utilizar as seguintes tecnologias: PLC, BPL, RF Mesh, entre outras.

A camada WAN é responsável por conectar os concentradores a distribuidora e para isso pode ser utilizado: fibra ótica, rede de celular ou PLC, por exemplo. Após

o envio através da WAN, os dados chegam na distribuidora onde serão utilizados nos mais diversos serviços.

O módulo MDMS (*meter data management system*) é um modulo central, que é responsável por gerenciar, armazenar e analisar os dados recebidos. Além disso, é responsável por fazer a conexão entre os dados recebidos e os subsistemas legados existentes nas distribuidoras.

2.4 TARIFA BRANCA

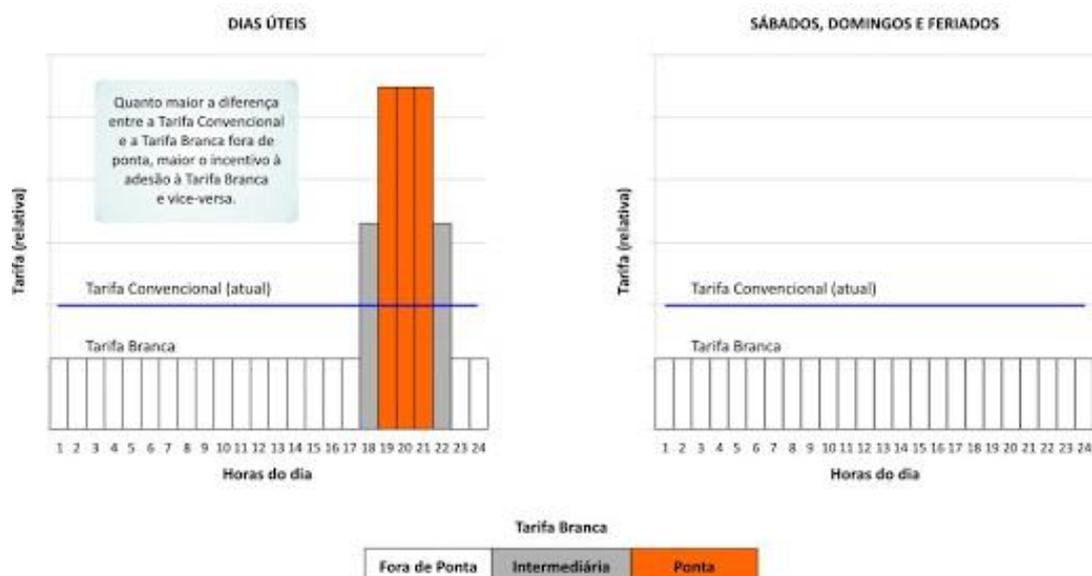
A tarifa branca é uma modalidade tarifária regulamentada pela ANEEL, que permite a precificação diferenciada da energia com relação ao horário e o dia da semana. Essa modalidade tarifária é oferecida para as unidades consumidoras do grupo B. [14]

Essa nova modalidade é possível graças aos medidores inteligentes de energia, pois esses são preparados para implementar múltiplos modelos tarifários. Além disso, essa modalidade se encaixa no conceito de *smart grid*, pois visa diminuir a demanda nos horários de pico da rede, distribuindo melhor a demanda.

Nessa modalidade de consumo há três possibilidades de preço pelo kWh, divididos em: ponta, intermediário e fora de ponta. O horário fora de ponta é o horário com a menor tarifa e abrange boa parte do dia. O horário intermediário, que compreende a hora anterior e a hora posterior ao horário de ponta, tem uma tarifa intermediária, servindo como uma espécie de transição. O horário de ponta compreende o período de pico do sistema, no período noturno, e tem a tarifa mais cara. Durante os finais de semana e feriados nacionais o dia inteiro é tarifado como fora de ponta.

Os valores tarifários são regulamentados para cada região e geralmente são revisados a cada quatro anos. Segue abaixo uma imagem que mostra como é a distribuição dos horários e uma comparação com o modelo convencional.

Figura 4 - Apresenta o funcionamento da tarifa branca durante os dias úteis (esquerda) e finais de semana/feriados. Além disso, é possível fazer uma comparação entre as tarifas existentes.



Fonte: ANEEL [29]

Essa modalidade tarifária visa que o consumidor crie o hábito de consumir a energia fora do período de pico da rede, fazendo com que o consumo da rede seja melhor distribuído. Essa tarifa pode ser vantajosa para alguns grupos de consumidores, sendo aconselhável antes de solicitar a mudança de modalidade uma análise dos hábitos de consumo.

3 PROTÓTIPO DESENVOLVIDO

3.1 REVISÃO TEÓRICA

3.1.1 UPM

O repositório UPM (*Useful Packages & Modules*) é um repositório público no GitHub que contém diversos drivers de sensores e atuadores implementados com a contribuição da comunidade. Esse repositório trabalha em conjunto com a API MRAA trazendo mais simplicidade na implementação de diversos sensores disponíveis no mercado.

O repositório dá suporte a uma quantidade gigantesca de sensores e atuadores, muito devido a massiva participação da comunidade. Esse repositório disponibiliza diversos códigos para a programação de sensores e atuadores em diversas linguagens de programação, entre elas C/C++, Java, Python e Node.JS.

Para o uso dos códigos basta a realização da clonagem ou o download do repositório na página do GitHub do projeto. Na web há disponível uma página, que contém toda a documentação e exemplos de implementação dos sensores disponíveis no repositório. [15]

3.1.2 LIBMRAA

A biblioteca LIBMRAA ou só MRAA é uma biblioteca feita em C/C++, e que pode ser utilizada em Java, Python e JavaScript, visando mapear as interfaces I/O de diversos hardwares existentes.

O principal objetivo dessa biblioteca é gerar um código de alto nível que não seja específico a uma única plataforma de hardware, onde a MRAA realiza o mapeamento das interfaces I/O e adapta o código de alto nível as especificidades da plataforma utilizada.

Esse processo é realizado durante a etapa de compilação, tirando a necessidade de o desenvolvedor conhecer a fundo a plataforma usada. Outra

vantagem dessa biblioteca é permitir uma fácil troca de plataforma de hardware sem a necessidade de grandes alterações no código de alto nível.

Essa biblioteca também é um projeto *open source* alocado no Github, bastando realizar a clonagem do repositório para a sua utilização. Além da página no GitHub, existe uma página específica com a documentação e exemplos de uso [16].

3.1.3 MQTT

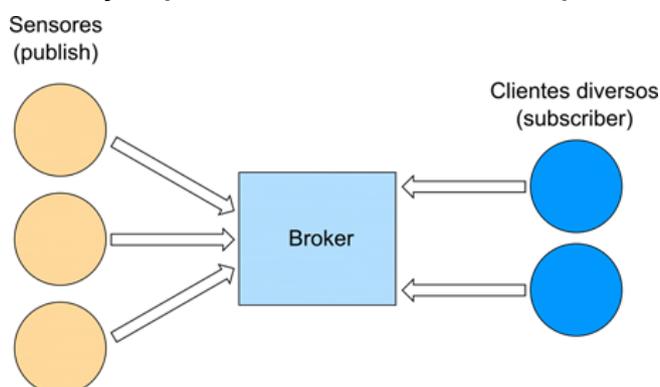
O protocolo MQTT (*Message Queue Telemetry Transport*) é um protocolo projetado para comunicação por troca de mensagens entre dispositivos que não disponham de alta largura de banda e que precisem de um modo confiável de comunicação [17].

Para implementações de aplicações IoT, esse protocolo é muito utilizado pela simplicidade de implementação, qualidade e segurança [18]. O protocolo utiliza o protocolo TCP/IP para se comunicar com a internet.

A comunicação utilizando esse protocolo se dá através de um padrão de troca de mensagens *publish/subscriber*, onde para a escrita ou leitura de informações de um determinado elemento da rede é necessário passar por um *broker*. O *broker* é um mediador entre o *publish* e o *subscriber* centralizando a comunicação e desacoplando as partes envolvidas na comunicação.

A imagem a seguir ilustra o que foi explicado acima.

Figura 5 - Ilustração que ilustra o funcionamento do protocolo MQTT.



Fonte: Wikipedia [30]

As mensagens são identificadas utilizando o conceito de tópicos, onde pode-se manipular a forma com que os dados são enviados ou recebidos, permitindo gerenciar como os dados serão enviados. Existe a possibilidade do uso de sistema de login e o uso de criptografia neste protocolo, além da possibilidade de configurar a qualidade do serviço de transporte das mensagens gerando robustez a rede.

O protocolo MQTT tem implementações em diversas linguagens de programação e tem distribuições abertas ou privadas.

3.1.4 SPI

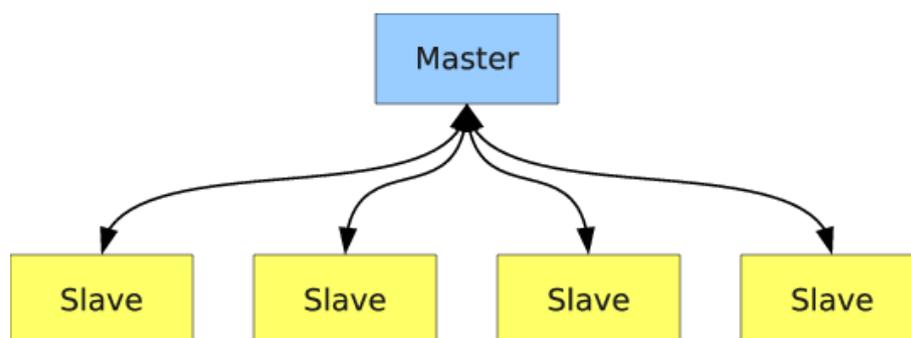
O protocolo SPI (*Serial Peripheral Interface*) é um protocolo de comunicação síncrono desenvolvido inicialmente pela Motorola, que é muito utilizado para a comunicação de circuitos integrados. É um protocolo de alta velocidade, *full duplex* e síncrono. [19]

A comunicação síncrona realizada por esse protocolo, utiliza um conceito conhecido como Mestre-Escravo, onde há:

- um dispositivo Mestre, que gera um sinal de *clock* e sinais de comando;
- e dispositivos que estão ligados a esse mestre, denominados escravos.

A seguir segue uma imagem que ilustra o conceito acima.

Figura 6 - Ilustração de uma comunicação SPI composta por diversos escravos.



Fonte: Embarcados [19]

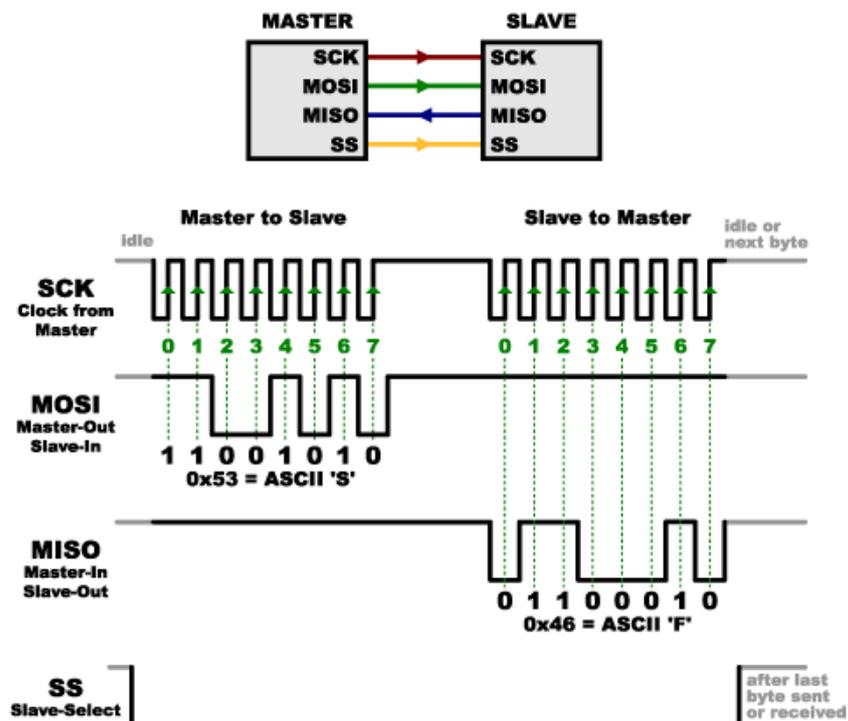
Fisicamente o protocolo é implementado usando 4 pinos, são eles:

- Chip Select ou CS, responsável por ativar ou desativar a comunicação entre o Mestre e o Escravo. Utiliza logica negativa para o controle, onde

quando o sinal estiver em 1 não há comunicação e quando estiver em 0 estabelecesse a comunicação entre o Mestre e o Escravo;

- Serial Clock ou SCLK, responsável pelo clock simétrico, que será utilizado para sincronizar o Mestre com seus Escravos;
- Serial Input ou SI, responsável por enviar os dados do Mestre para o Escravo;
- Serial Output ou SO, responsável por enviar os dados do Escravo para o Mestre.

Figura 7 - Ilustração do funcionamento da comunicação SPI.



Fonte: PROMETEC [31]

A comunicação é feita de forma *full-duplex*, onde toda a troca de dados acontece sempre em ambas as direções. A taxa de transmissão é alta podendo ultrapassar 10Mbps. A saída do sinal da interface segue o esquema *push-pull* e pode-se configurar as bordas de comunicação do *clock* através da polaridade e da fase, gerando 4 modos de uso.

Para um melhor entendimento de como se dar a transmissão e o formato de envio e recebimento dos dados, sempre vale consultar o *datasheet* do CI e olhar os

gráficos de transmissão (que possibilitam descobrir o modo de uso, se está configurado como MSB ou o LSB e o *clock* recomendado de funcionamento).

3.2 HARDWARE

3.2.1 CS5463

Informações básicas

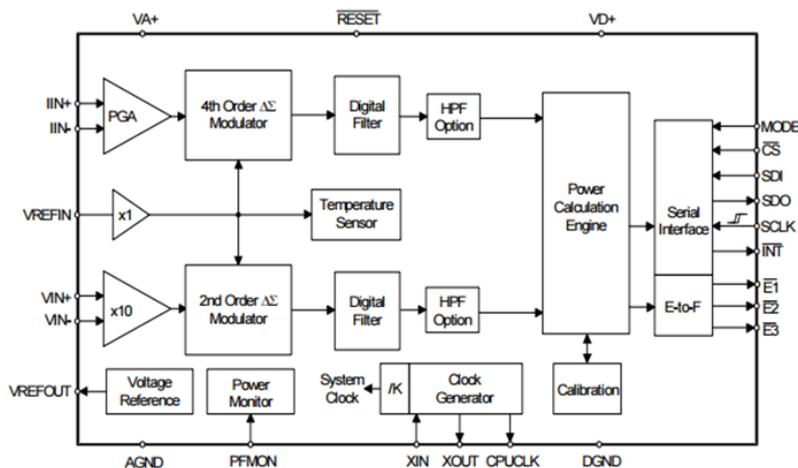
O circuito integrado CS5463 é um CI empregado na implementação de medidores digitais de energia de forma precisa, bastante linear e econômica. Esse CI é fabricado pela Cirrus Logic e permite medições de diversas grandezas elétricas em regime AC [20].

O chip permite obter diversas informações, onde vale destacar as medições de tensão e corrente instantânea e RMS, potência ativa, reativa e aparente, fator de potência e frequência de rede. Além disso, o chip possibilita calibração interna e medição de temperatura interna do CI.

O CI combina um módulo de cálculo, dois ADCs do tipo Sigma Delta e uma interface serial em um único circuito. É projetado para utilizar uma gama de sensores de tensão e de corrente. Para a medição de tensão pode-se utilizar transformadores de tensão ou divisores resistivos. Para a medição de corrente pode-se utilizar transformadores de corrente, resistor *shunt* ou sensor *hall*.

Segue o diagrama interno do CI:

Figura 8 - Diagrama interno do chip.

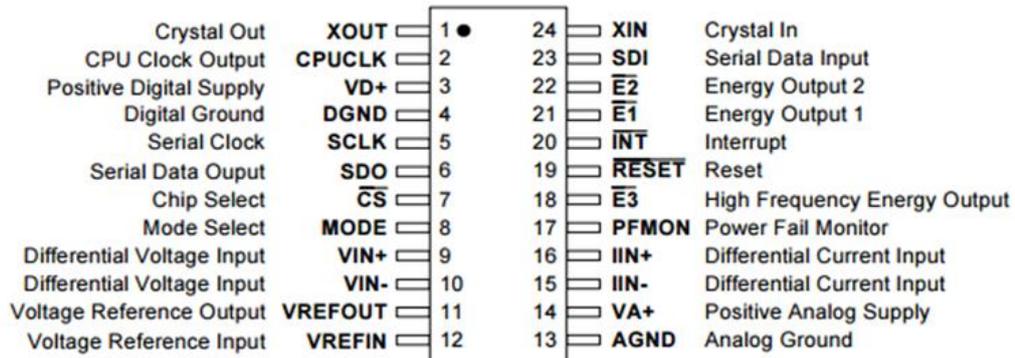


Fonte: Cirrus Logic [20]

Características físicas

O chip é encapsulado no formato TSOP de 24 pinos. Basicamente pode-se dividir os pinos em: pinos relacionados ao *clock*, pinos relacionados à comunicação serial, pinos de entrada e saída de sinais analógicos e pinos de alimentação.

Figura 9 - Diagrama dos pinos do CI.



Fonte: Cirrus Logic [20]

O CI é dividido em dois setores de alimentação (analógico e digital), que são alimentados separadamente. A parte digital, VD+ pode ser alimentada com 3,3V ou 5V. A parte analógica, VA+ deve ser alimentada com 5V.

Há a necessidade de manter uma tensão de referência no valor de 2,5V. O CI necessita de um *clock* para funcionar sendo recomendado o valor de 4,096 MHz. O CI se comunica serialmente utilizando o protocolo SPI, com *clock* máximo de 2MHz - o projeto utiliza um *clock* de 1MHz.

As entradas de sinais de medição, tanto para tensão, quanto para corrente, são feitas de forma diferencial, visando diminuir o ruído. No caso do sinal de tensão a máxima tensão admitida na entrada é de 500 mV pico a pico. No caso do sinal de corrente há a possibilidade de se utilizar como tensão de entrada 100 ou 500 mV pico a pico dependendo da programação interna do CI, que no caso utiliza um amplificador operacional de ganho programável. Internamente o CI tem implementado filtros IIR e filtro passa-alta.

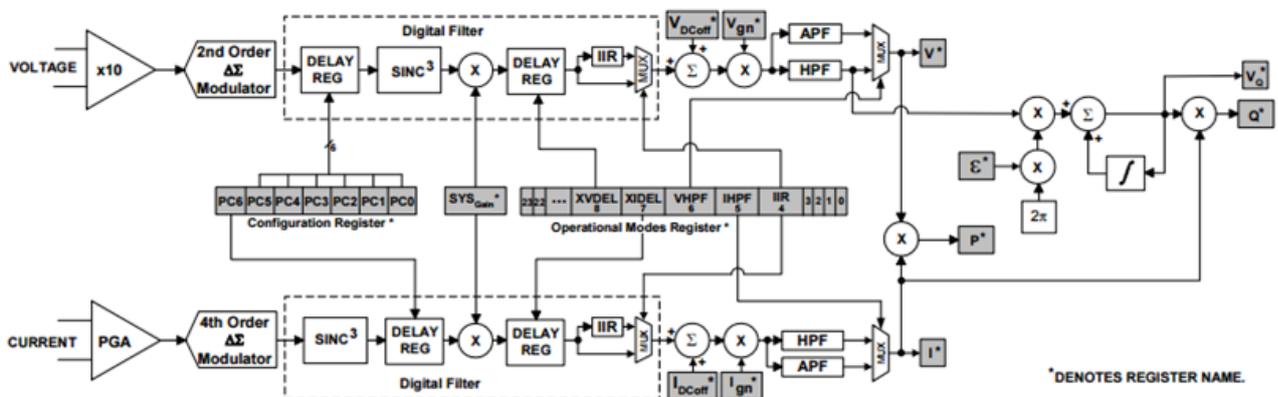
O *chip* tem alguns pinos de saída programáveis e a possibilidade de realizar interrupções bastando enviar corretamente os comandos via SPI (mais informações no *datasheet*) [20].

Funcionamento básico

Abaixo, seguem os diagramas de funcionamento do CI CS5463 retirados do *datasheet*, mostrando como são executados os cálculos dos diversos valores calculados pelo CI. A primeira imagem, mostra como é feita o cálculo das grandezas elétricas. Basicamente as duas entradas de medição (corrente e tensão), passam por amplificadores operacionais e são processadas pelo ADC Sigma Delta. Após isso, os valores obtidos passam pelos filtros digitais, que em conjunto com algumas informações contidas em registradores iniciam os ajustes dos valores para os cálculos de tensão e corrente. Depois, é feito o cálculo dos valores de potências, tensão e épsilon (que permite calcular a frequência da rede).

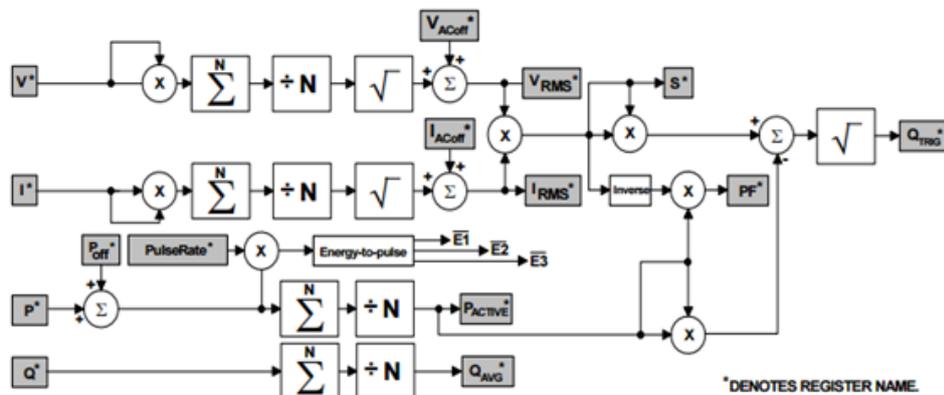
Passada essa parte inicial são realizados os demais cálculos, seguindo as etapas mostradas na segunda imagem.

Figura 10 - Diagrama de funcionamento lógico 1.



Fonte: Cirrus Logic [20]

Figura 11 - Diagrama de funcionamento lógico 2.



Fonte: Cirrus Logic [20]

Comunicação serial, comandos internos e acesso de registradores

O CS5463 utiliza o protocolo SPI para se comunicar. Como já explicado anteriormente, este protocolo utiliza quatro pinos de comunicação e no caso do CS5463 ele funciona com *clock* máximo de 2 MHz.

Para estabelecer a comunicação serial é necessário reiniciar o chip para sincronizar o *clock*. Essa operação segue os seguintes passos:

- Ativar o pino CS (pino utiliza lógica negativa), que está ligado à porta do microcontrolador/microprocessador;
- Ativar o pino de *Reset* (pino utiliza lógica negativa), que está ligado à uma porta do microcontrolador/microprocessador durante no mínimo 10 microssegundos;
- Enviar pela interface SPI por três vezes os comandos 0xFF e uma vez o comando 0xFE para finalizar a sincronização dos dispositivos.

Terminado o processo de estabelecer a comunicação entre os dispositivos, é possível realizar uma comunicação bidirecional entre as partes. O *chip* permite realizar a configuração de diversos registradores que influenciam nos valores calculados pelo CI, enviar diversos comandos que influenciam na forma como funciona o *chip* (*reset*, resincronização, formato de coleta de dados e modos de consumo de energia) e enviar comandos de leitura e escrita aos registradores, sendo que existe uma grande quantidade de comandos possíveis.

Abaixo, segue o formato utilizado para um comando.

Figura 12 - Formato de um comando de escrita ou leitura.

B7	B6	B5	B4	B3	B2	B1	B0
0	W/R	RA4	RA3	RA2	RA1	RA0	0

Fonte: Cirrus Logic [20]

Um comando tem o tamanho de 1 *byte* e dependendo dos *bits* configurados realiza-se umas das operações descritas no *datasheet*. Quando são utilizados os comandos de leitura, é necessário o envio de uma espécie de pacote, onde se envia o comando de leitura e três comandos com o valor 0xFF. Quando são usados os comandos de escrita é necessário enviar a mesma espécie de pacote, porém o primeiro byte tem o comando de escrita e os 3 bytes contém o conteúdo a ser escrito.

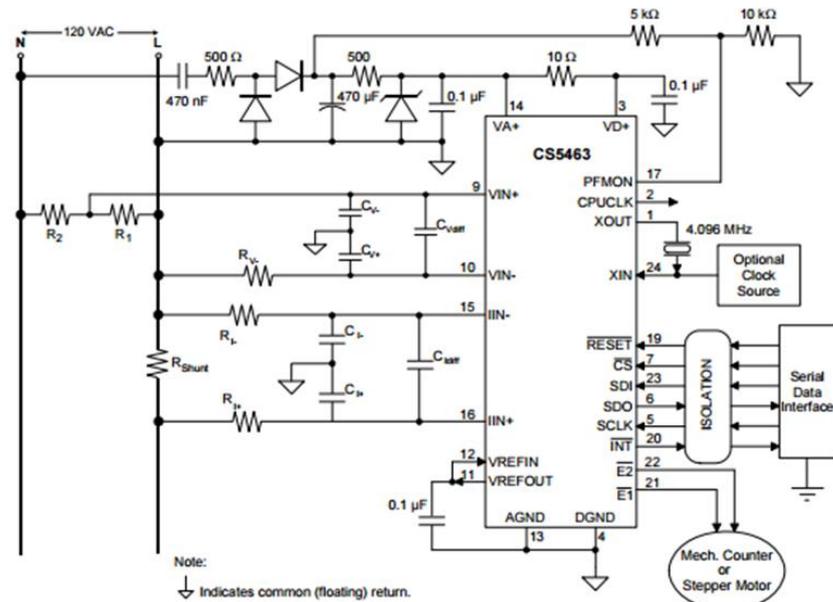
Os comandos de leitura ou escrita de registradores possibilitam o acesso a uma grande quantidade de dados e configurações. Existe no CI três páginas de registradores, onde para mais informações olhar o *datasheet* [20]. O valor obtido nestes registradores é proporcional aos valores reais da rede, sendo que na sua maioria assumem o formato de 0 a 1 ou -1 a 1. Para realização da conversão dos valores proporcionais em valores “reais” foi utilizado: as informações do *datasheet* [20] e o documento de design da Cirrus Logic relativo ao CI [21].

Arquiteturas de hardware

Os documentos da Cirrus Logic [20] [22] apresentam dois diagramas de conexão para circuitos monofásicos.

O primeiro diagrama tem componentes de baixo custo, sem isolamento da rede alternada e utiliza um divisor de tensão e um resistor *shunt* (resistência baixíssima) como elementos sensores. Além disso, é mostrada uma possível fonte de alimentação, o valor do cristal de *clock* necessário para o CI e os pinos que são ligados ao microcontrolador (com o adendo dá necessidade de um circuito de isolamento).

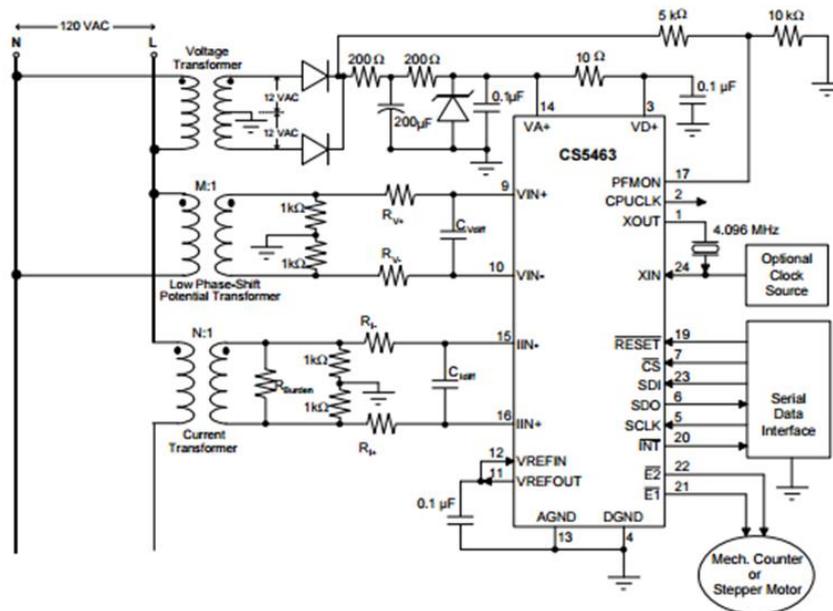
Figura 13 - Diagrama de conexão, monofásica e conexão direta com a rede de energia.



Fonte: Cirrus Logic [20]

O diagrama abaixo - utilizado no protótipo desenvolvido - conta com isolamento da rede alternada, devido ao uso de transformadores como elementos sensores. Os sensores utilizados nesse diagrama, são mais caros comparados ao diagrama anterior, porém com um risco menor de queima total dos componentes. As demais ligações são semelhantes ao diagrama anterior.

Figura 14 - Diagrama de conexão, monofásico com isolamento.



Fonte: Cirrus Logic [20]

3.2.2 Intel Edison

A placa Intel Edison é uma placa de desenvolvimento da Intel, que tem em seu SoC um processador Atom *Dual Core* e um microcontrolador Quark, tendo respectivamente 500 MHz e 100 MHz de processamento. A placa tem 1 GB de memória DDR3, 40 GPIOs, *Bluetooth* e *Wifi* embutido. A placa utiliza uma distribuição Linux, modificada e gerada usando o Yocto [23].

A placa Intel Edison é de tamanho reduzido, necessitando a utilização de placas adaptadoras para se ter o acesso as portas GPIOs. No projeto, foi utilizado uma *Breakout Arduino*, imagem a seguir, que possibilita o acesso as portas GPIOs, num formato seguindo a pinagem do Arduino Uno. Isso é interessante, pois permite o uso de diversos módulos (focados em Arduino) existentes no mercado.

Figura 15 - Intel Edison e a placa *Breakout Arduino*.



Fonte: MAKER PRO [32]

Na imagem da *Breakout Arduino*, é possível ver o leitor de SD, jack de alimentação e os pinos de acesso no formato do Arduino Uno R3.

3.2.3 Sensor de corrente

O sensor de corrente utilizado no protótipo foi um YHDC SCT-013-000 da empresa chinesa Dechang Electronics. Esse sensor é um transformador de corrente de núcleo dividido, não intrusivo. Este sensor permite a medição de correntes de até 100 A, tem diodo interno para proteção de corrente reversa e seu secundário fornece uma saída de corrente de até 50 mA, onde é necessário colocar um resistor de carga para gerar uma tensão proporcional.

Pontos fortes desse sensor são:

- Isolamento galvânico entre o circuito medidor e a rede elétrica;
- Fácil instalação, não havendo a necessidade de romper o fio e fazer uma emenda para conseguir instalar o elemento sensor;
- Não existe a necessidade de alimentação auxiliar para o funcionar;
- Circuito de condicionamento simples (basicamente um resistor de carga e um filtro).

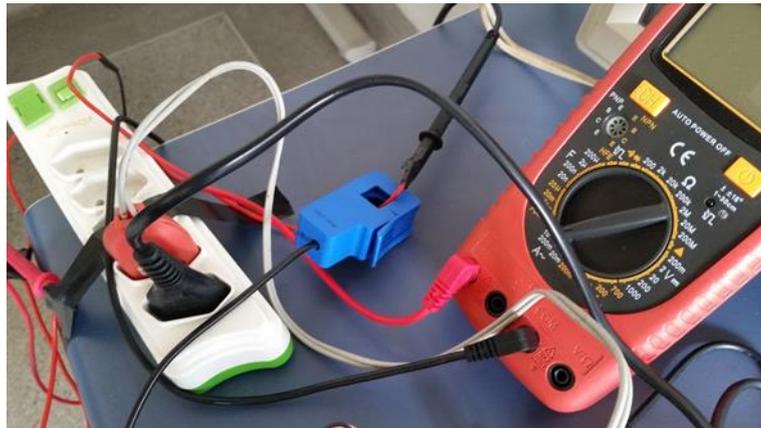
A seguir, segue duas imagens do sensor. Uma mostrando o sensor separadamente e outra com o sensor instalado.

Figura 16 - Transformador de corrente YHDC SCT-013-000.



Fonte: FILIPEFLOP [33]

Figura 17 - Sensor instalado.

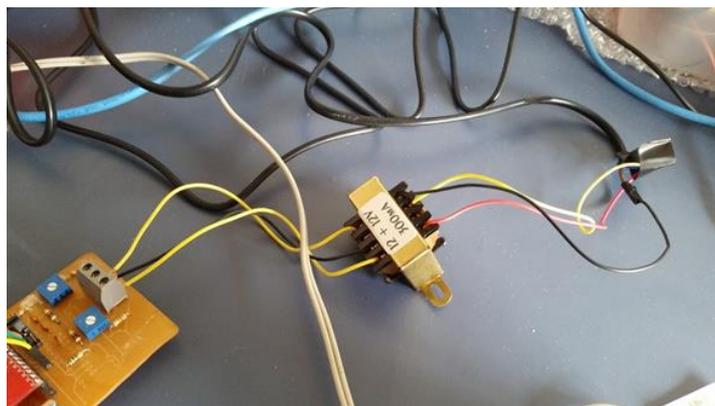


Fonte: Próprio autor

3.2.4 Sensor de tensão

O sensor de tensão utilizado foi um transformador de tensão 12 + 12 V 300mA, que em conjunto com um circuito divisor de tensão, gera a tensão proporcional de no máximo 250mV pico. Este transformador é de fácil aquisição e manipulação, permitindo que não haja uma ligação direta entre o protótipo e a rede, algo necessário para minimizar o risco de queima dos circuitos. Abaixo, segue uma imagem do transformador de tensão instalado e funcionando.

Figura 18 - Transformador MAXXI 12 + 12 V 300mA utilizado.



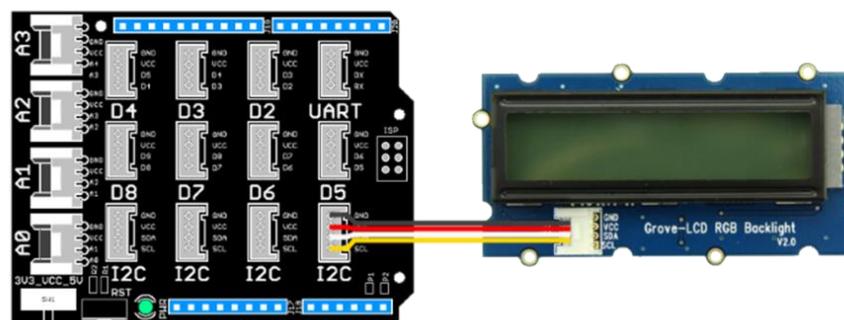
Fonte: Próprio autor

3.2.5 LCD e Base Shield

Para apresentar os resultados localmente foi utilizado um display LCD RGB Backlight e uma Base Shield desenvolvidos pela Grove. O display LCD utilizado é um display de 16 x 2 (16 colunas por 2 linhas), que pode ter sua cor de fundo modificada devido ao LED RGB incluso. Esse display tem embutido um modulo de interface I2C, permitindo uma considerável redução na quantidade de pinos usados para o funcionamento do display e uma considerável simplificação no software para realizar a programação do display.

A Base Shield serviu para realizar a conexão do display com a placa, pois já vem com o plug e o cabo específico para ligação, diminuindo erros de ligação e mau contato.

Figura 19 - Display LCD RGB I2C e Base Shield.



Fonte: AppMarkers [34]

3.3 ARQUITETURA PROPOSTA

3.3.1 Hardware

Para a implementação do protótipo foi necessário analisar diversos *datasheets* e arquivos de referência.

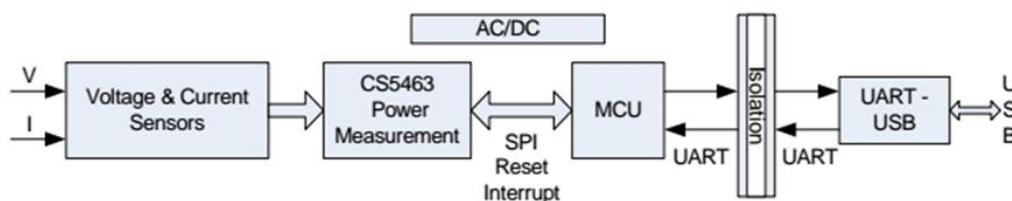
A parte de processamento não demandou escolha, pois foi obtida de forma gratuita. A placa de desenvolvimento tem um microprocessador Atom, acesso à Internet via Wifi, Sistema Operacional Linux e interface SPI, se encaixando nas necessidades do projeto. Essa placa tem como contras: o seu elevado custo no Brasil

- por isso a importância da biblioteca MRAA para o projeto, e a placa foi descontinuada pela Intel.

Visando escolher o *chip* de medição e os sensores que se encaixam no cenário proposto – baixo custo e fácil obtenção, analisou-se os arquivos de design da Cirrus Logic, Analog Devices e ST, onde foi possível encontrar diversas similaridades entre as arquiteturas propostas por esses fabricantes.

As duas imagens a seguir ilustram as arquiteturas de hardware propostas pela Cirrus Logic e pela ST. A arquitetura proposta pela Cirrus Logic é mais simples, pois a empresa só disponibiliza soluções para microprocessamento e medição especializada. A arquitetura da ST apresenta mais elementos e funcionalidades, pois a empresa tem uma linha focada em *smart grid*.

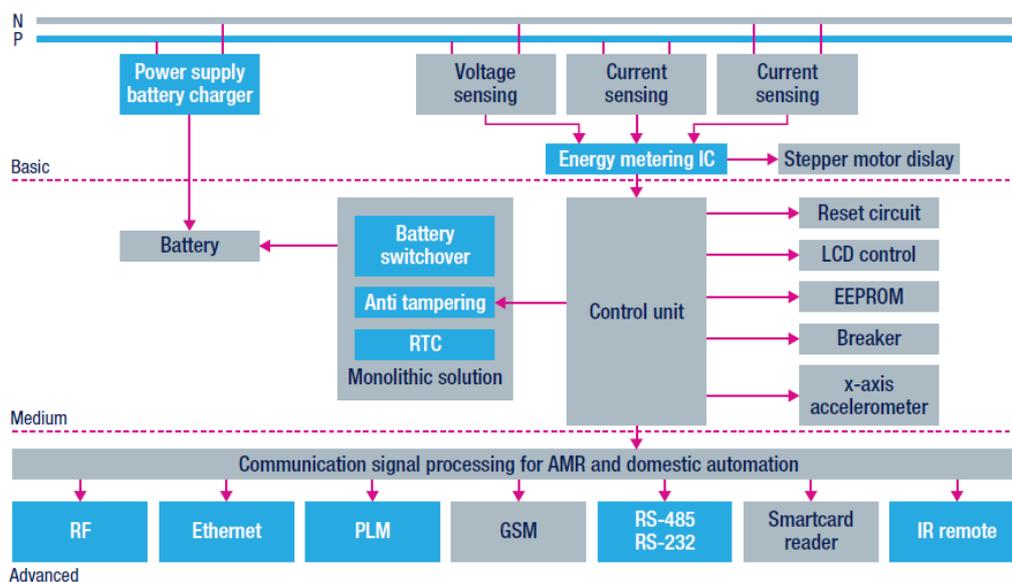
Figura 20 - Arquitetura proposta pela Cirrus Logic no documento de design.



Fonte: Cirrus Logic [22]

Figura 21 - Diagrama de blocos - Medidor Inteligente - ST.

SMART E-METER BLOCK DIAGRAM



Fonte: ST [35]

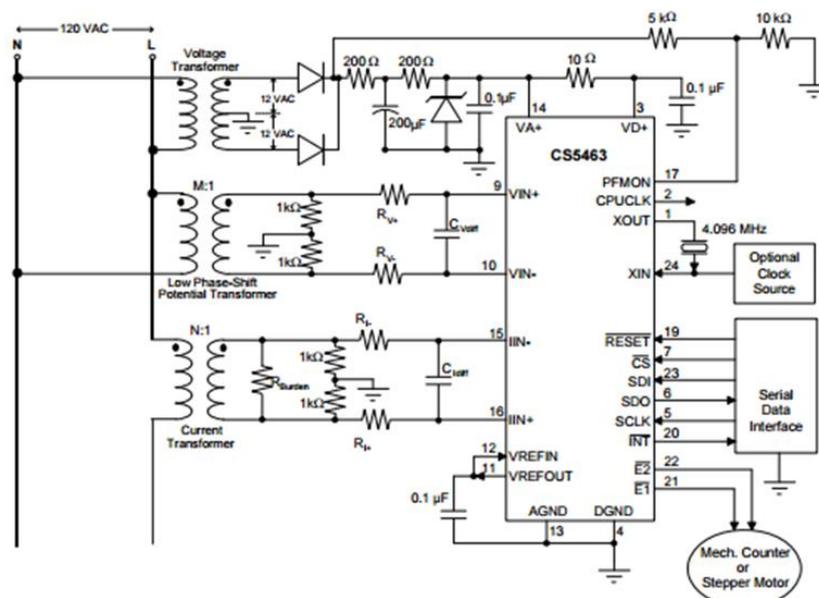
Para o protótipo foi escolhido o CI CS5463. Esse *chip* é de fácil aquisição, devido ao seu baixo custo, custando por volta de U\$ 1,50. O CI tem fácil comunicação, diversas funcionalidades in hardware e é possível conseguir algumas unidades gratuitas com a fabricante para teste (o que foi o caso).

O *chip* permite a realização de diversas medições e tem presente rotinas de calibração, simplificando o desenvolvimento e minimizando possíveis erros de implementação de software. O CI utiliza interface SPI para a comunicação com a MCU, algo interessante pois a maioria dos microcontroladores tem esse protocolo implementado. Por fim, há uma grande quantidade de pessoas utilizando ou tentando utilizar esse chip, gerando uma comunidade que produz muito material a respeito.

Com o CI escolhido, analisou-se o *datasheet* e os arquivos de referência disponíveis no site na Cirrus Logic. Esse CI tem 4 formas de uso previsto, onde foi escolhida a forma abaixo. Esse diagrama prevê o uso em instalações monofásica e utiliza como elemento sensor um transformador de potência (TP) e um transformador de corrente (TC). O uso de TC e TP tem como vantagens:

- Não acoplamento do medidor a rede alternada, diminuindo o risco de queima total do sistema;
- Ser menos intrusivo;
- Fácil de instalar.

Figura 22 - Diagrama de conexão escolhido.



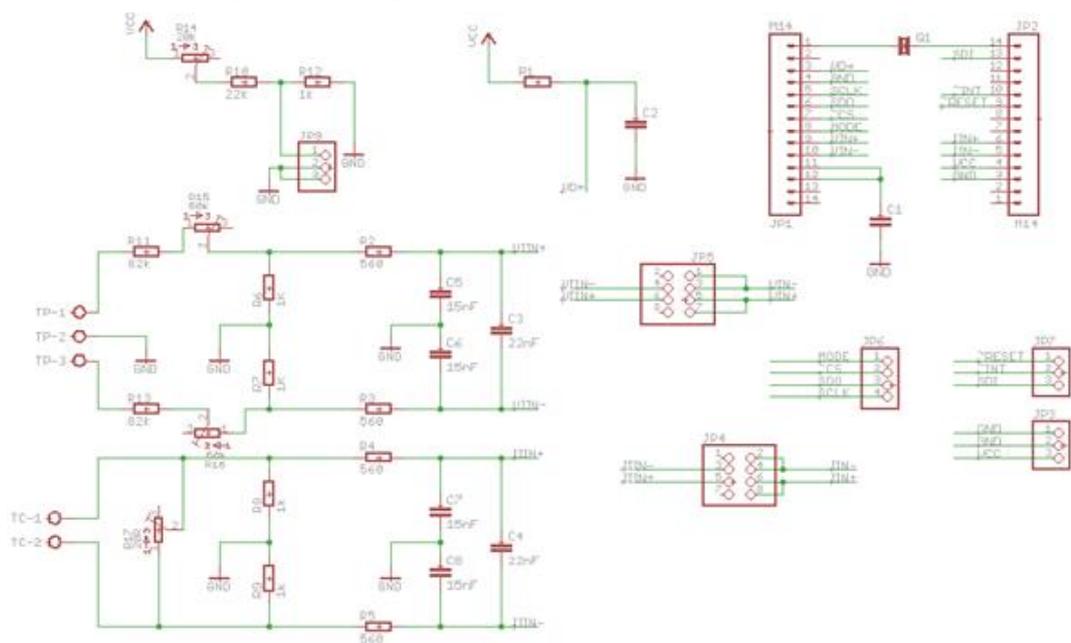
Fonte: Cirrus Logic [20]

Com o CI medidor e os sensores escolhidos foi necessário projetar e confeccionar uma placa PCB, visando implementar o diagrama da Figura 22. A placa foi desenvolvida com as seguintes premissas:

- Gerar a interface de conexão entre os sensores e o CI medidor através de bornes e jumpers;
- Gerar a interface física para a conexão entre o CI CS5463 e a Intel Edison - ou qualquer placa de desenvolvimento que tenha protocolo SPI e trabalhe com 5V - para o uso do protocolo SPI;
- Prover os circuitos necessários para o bom funcionamento da aplicação, no que tange ao condicionamento dos sinais dos sensores e dos componentes necessários para o CI funcionar conforme o *datasheet*;
- Possibilitar a ligação de pontas de testes a partir de pontos facilitados, buscando facilitar a aferição das medidas.

Abaixo, segue o diagrama da placa projetada. Ela foi desenhada utilizando o software Eagle. O diagrama elétrico seguiu o que está descrito no *datasheet* e na imagem 22. Foram realizados os cálculos das resistências responsáveis pelo condicionamento do sinal e a simulação desses circuitos no Proteus. Para uma melhor precisão foram utilizados *trimpots*, buscando alcançar um valor de resistência próximo do calculado.

Figura 23 - Diagrama do circuito impresso (protótipo).



Fonte: Próprio autor

O cálculo dos resistores que servem para acondicionar o valor da tensão foi feito utilizando a fórmula a seguir (divisor de tensão):

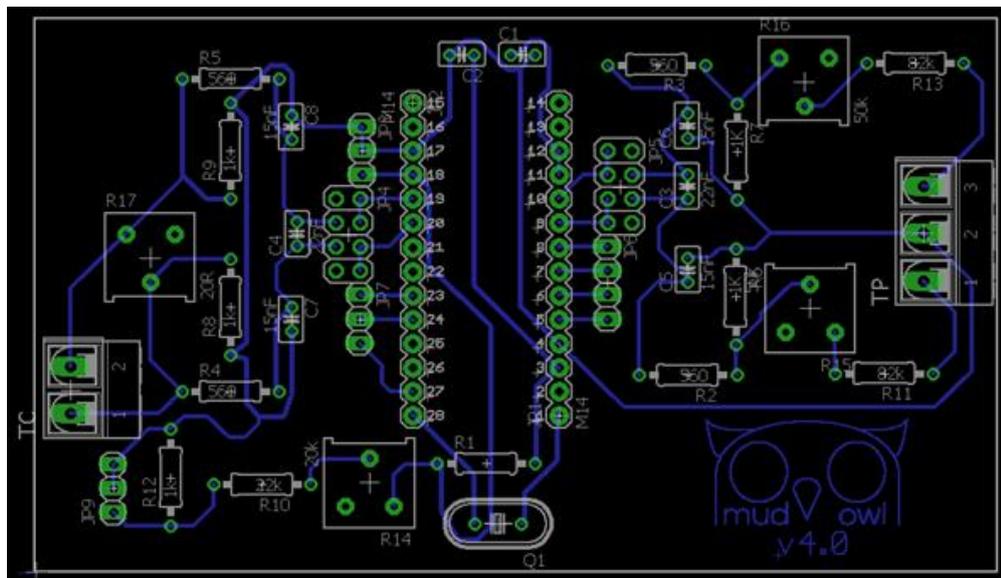
$$V_{out} = (R2 / (R1 + R2)) * V_{in}$$

O cálculo do resistor de carga utilizado pelo transformador de corrente, utilizou a fórmula a seguir (Lei de Ohm):

$$V = RI$$

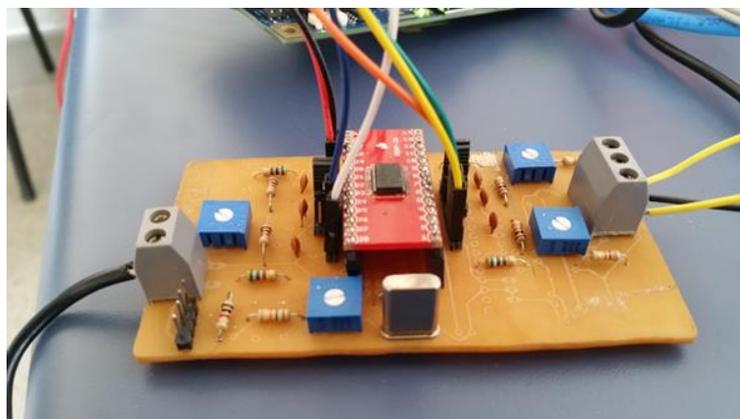
Com o diagrama feito, foi possível desenhar o esquemático da placa, que foi posteriormente utilizada na manufatura da placa (PCB). Abaixo, segue o esquemático do protótipo e a placa finalizada.

Figura 24 - Esquemático final.



Fonte: Próprio autor

Figura 25 - Placa finalizada.



Fonte: Próprio autor

Por fim, foi escolhido para mostrar as informações localmente um display LCD I2C fabricado pela Grove. Este display tem como diferencial o uso do protocolo I2C para a realização da comunicação entre o display e a placa de desenvolvimento.

3.3.2 Software

Antes de introduzir as questões gerais do software é importante fazer algumas considerações a respeito da placa de desenvolvimento.

A Intel Edison vem de fábrica com uma versão compacta do Linux, podendo ser otimizada dependendo da aplicação. O SO já vem instalado com compilador C/C++ e interpretadores Python, Java e Node.js. Durante o projeto foi utilizado no desenvolvimento do *firmware*, Python 2.7 e as bibliotecas: *datetime*, *upm*, *mraa*, *csv*, *json*, *os*, *paho* e *math*.

Vale o destaque a biblioteca MRAA, pois ela permite a portabilidade do código desenvolvido para outras placas que contenham Linux embarcado. Isso possibilita, por exemplo, o uso desse código numa plataforma mais barata.

Os softwares desenvolvidos visam implementar:

- Firmware:
 - geração da comunicação SPI entre o *chip* e a placa embarcada;
 - implementação do processamento dos dados recebidos do *chip*;
 - protocolo MQTT para comunicação entre a placa e a *dashboard*.
 - implementação da visualização num LCD com dados locais;
- Visualização online:
 - protocolo MQTT para comunicação entre a placa e a *dashboard*;
 - implementação da *dashboard*.
- *Plus*: implementação de uma versão simplificada da tarifa branca.

Firmware

Com o hardware do protótipo desenvolvido há a necessidade de lidar com as portas GPIOs e implementar a interface de comunicação SPI, para a comunicação

com o chip de medição. No projeto, utilizou-se a biblioteca MRAA para controlar as portas I/O e implementar a comunicação SPI.

Os códigos de implementação estão acessíveis no link:

https://bitbucket.org/Mud_Owl/ic_mud_owl_v2/src/master/

O trecho de código que estabelece a comunicação serial, basicamente consiste no estabelecimento de um *clock*, juntamente com a configuração do modo de operação e do *bit* mais significativo do sinal recebido. Estabelecido isso, basta controlar a porta que habilita ou desabilita a comunicação, possibilitando enviar ou receber *bytes*.

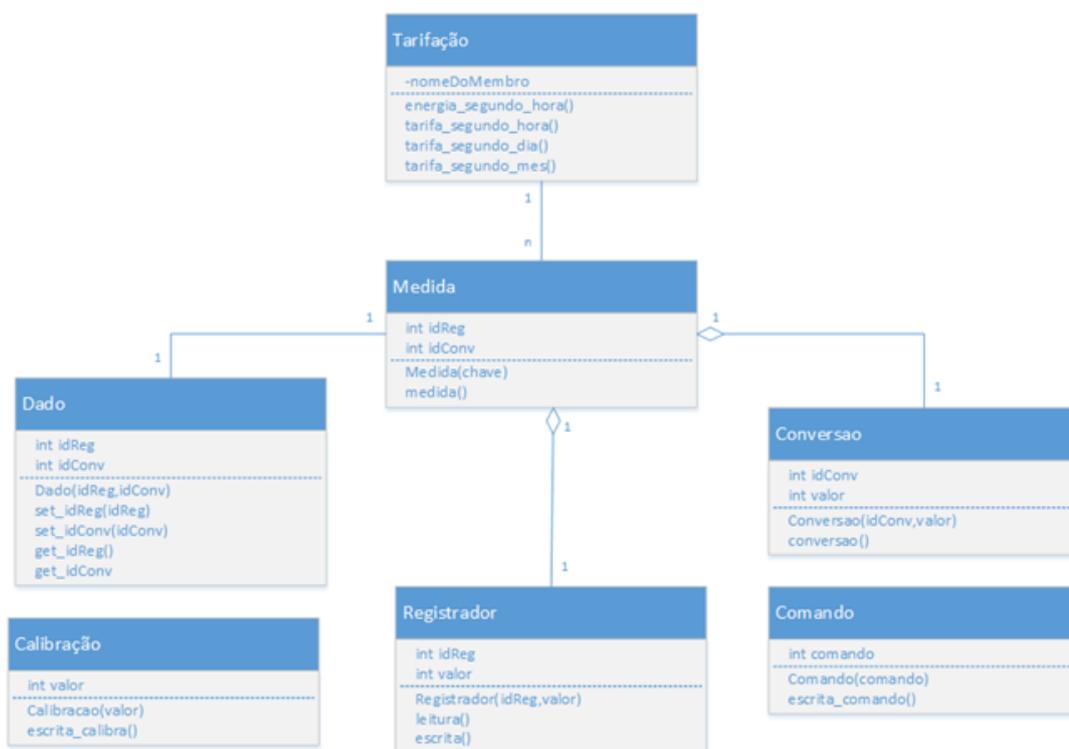
A comunicação é do tipo *full-duplex* e utiliza 4 pinos (um pino que estabelece o clock, um pino para controle da interface, um pino que recebe as informações do escravo e um pino que envia informações para o escravo). No caso do CS5463 para o envio de informação é necessário enviar 4 *bytes*, sendo 1 *byte* o comando e 3 *bytes* com FF de conteúdo. Para receber as informações são 4 *bytes*, sendo 1 do comando e 3 *bytes* com a informação propriamente dita.

Após o recebimento da informação, há a necessidade de concatenar os *bytes*, realizar algumas conversões e aplicar as constantes de escala.

A estrutura do *firmware* buscou seguir o paradigma orientado a objeto, onde se criou algumas classes para executar determinadas funções. São elas:

- classe “registrador”, cuida especificamente do acesso ou envio das informações para os registradores.
- classe “conversão”, cuida das conversões.
- classe “dados”, agrupa os comandos de leitura e escrita.
- classe “comandos”, agrupa os comandos de uso geral.
- classe “medida”, orquestra todas as funções, mostrando a medição pedida
- classe “Tarifação”, cuida das rotinas de tarifação.

Figura 26 - Diagrama de classes.



Fonte: Próprio autor

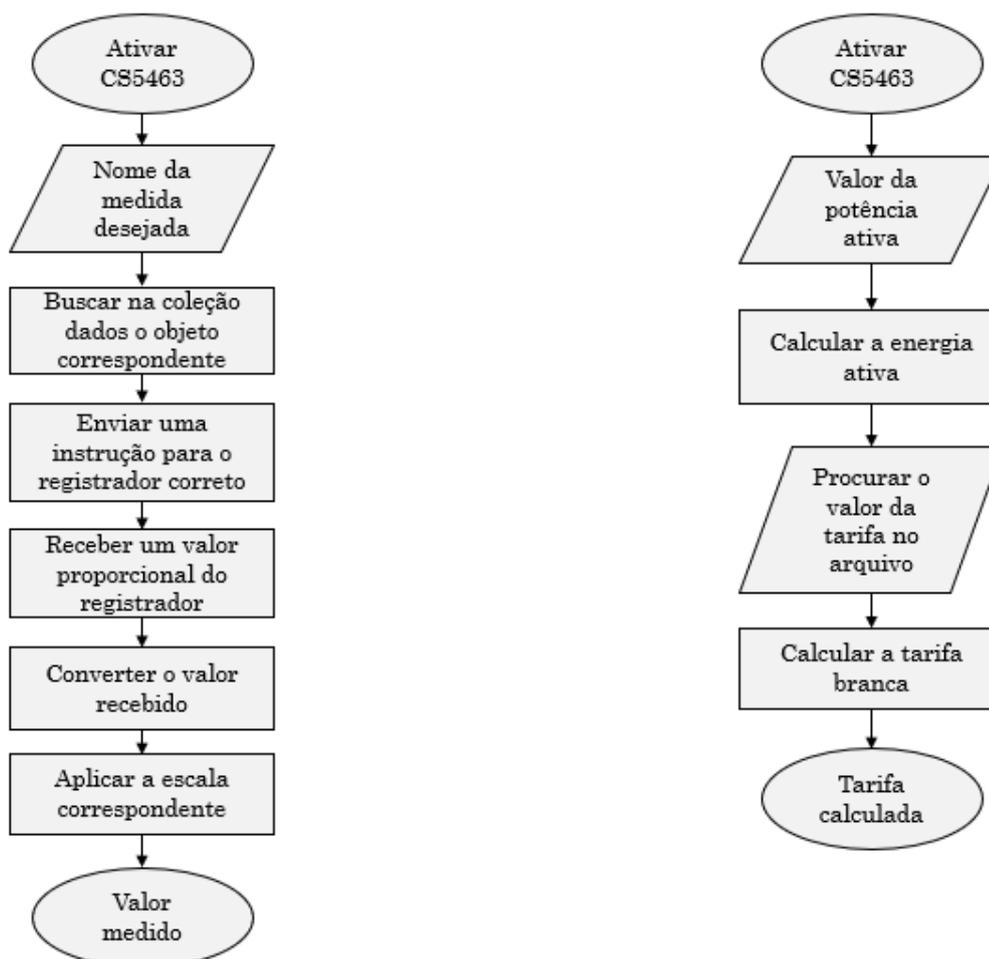
Passada a parte de processamento, é possível salvar as medidas no formato CSV e JSON, possibilitando analisar os dados posteriormente ou usar como memória de massa. O formato JSON é usado também no enviado de dados pelo protocolo MQTT.

Com o valor da potência ativa e o tempo obtido do sistema, é possível calcular o consumo de energia e realizar a tarifação.

O algoritmo de tarifação consiste em calcular o valor da energia consumida e aplicar a tarifa, distinguindo as bandeiras tarifárias baseadas nos dias da semana e no horário – seguindo o explicado na seção “Tarifa Branca”. O acesso as informações de data e hora se dá pela Intel Edison (RTC interno) e a biblioteca Datetime (escrita em Python).

A seguir, segue um fluxograma que ilustra os processos de medição e tarifação explicados anteriormente.

Figura 27 - Fluxogramas explicando como realizar a medição de uma grandeza elétrica e como realizar o cálculo da tarifa.



Fonte: Próprio autor

A implementação do cliente MQTT, que estabelece a comunicação com a *dashboard*, foi feita utilizando a biblioteca paho, escrita em Python, que contém as funções necessárias para a implementação do cliente MQTT.

Basicamente estabeleceu-se a comunicação e o envio de um arquivo JSON, com as informações a partir de um tópico MQTT. O servidor utilizado para a comunicação foi o "IoT Eclipse", porta 80, mantido pela fundação Eclipse.

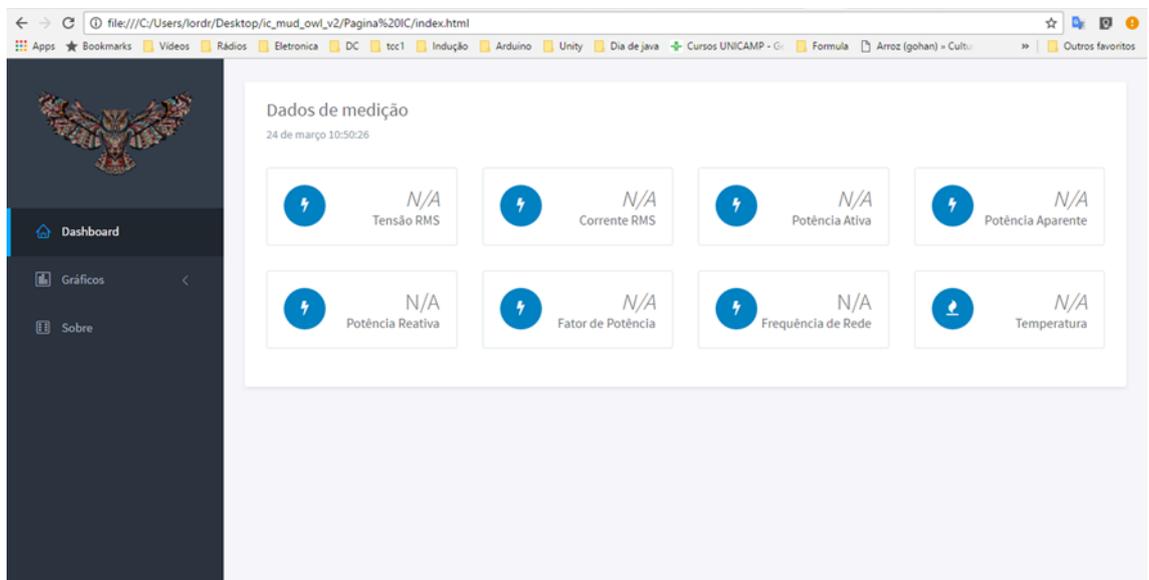
Por fim, o código que implementa a visualização no LCD, foi escrito em Python, e é baseado na documentação do fabricante do dispositivo, utilizando as funções/biblioteca do repositório UPM.

Visualização online

O cliente MQTT que cuida da comunicação entre o broker e a dashboard, foi escrita utilizando Javascript e a biblioteca paho. Basicamente, recebe-se o arquivo JSON contido no tópico e realiza um *parsing* da informação para o uso na dashboard.

As páginas que compõem o dashboard foram escritas utilizando HTML, CSS e Javascript, tendo como base um código gratuito distribuído pela themeined, com o nome Klorofil [24]. Para a plotagem dos gráficos foi utilizada uma API disponibilizada pela Highcharts [25], escrita em Javascript. Abaixo segue uma imagem, que mostra a dashboard desenvolvida.

Figura 28 - Página inicial da dashboard.



Fonte: Próprio autor

4 RESULTADOS E DISCUSSÃO

Durante todo o projeto houve uma preocupação em desenvolver um protótipo de baixo custo, para uso em residências e pequenos comércios. Tendo isso como premissa, buscou-se utilizar componentes de baixo custo, que estão listados na tabela a seguir. Essa tabela apresenta o preço mais baixo dos itens utilizados, inclusive dos que foram obtidos via amostra grátis (CS5463) e doação (Intel Edison).

Tabela 1 - Tabela de componentes e seus preços aproximados.

Itens	Preço Aproximado
Trimpots	R\$ 7,60
CI medidor CS5463	R\$ 1,80
Placa adaptadora	R\$ 8,50
Bornes	R\$ 3,00
Cristal Oscilador	R\$ 1,00
Resistores	R\$ 1,00
Capacitores	R\$ 2,40
Barras de pinos	R\$ 5,90
Placa de fenolite	R\$ 3,80
Jumpers	R\$ 10,00
Placa de desenvolvimento*	R\$ 200,00
Total	R\$ 245,00 / U\$ 45,00

Fonte: Próprio autor

Para a utilização do protótipo é necessário realizar a calibração nos *trimpots* do circuito de condicionamento. Para isso, ajustar-se os *trimpots*, com o CI desconectado visando alcançar a tensão máxima aceitável (máx 250 mV pico a pico). Após isso, conecta-se o CI no soquete e realiza-se as operações de calibração descritas no *datasheet* do CS5463 [20]. Além disso, é necessário que haja um ajuste criterioso nas escalas de conversão do software – isso envolve acertar os valores do arquivo JSON.

Finalizada a etapa de calibração, foi realizado um teste visando provar que o firmware desenvolvido é funcional. Para isso, decidiu-se medir e analisar o erro nas medições de tensão e corrente. Escolheu-se esse caminho, pois corrente e tensão

são as grandezas fundamentais, e as demais grandezas são calculadas com base nelas.

O teste seguiu a seguinte metodologia:

- Numa bancada, utilizando um filtro de linha adaptado, foram ligados o medidor e um conjunto de multímetros configurados como amperímetro e voltímetro;
- Escolheu-se um conjunto de equipamentos elétricos, no caso uma lâmpada incandescente de 25 W (representando os equipamentos com cargas resistivas), um ferro de passar roupa de 1200 W (equipamento com alto consumo de corrente) e uma bateadeira de 350 W (representando os equipamentos com cargas reativas);
- Ligou-se um equipamento de cada vez e realizou-se a coleta dos dados obtidos tanto pelo medidor, quanto pelos multímetros. Foram coletadas 690 amostras em um arquivo CSV e calculou-se a média (usando um Excel) para obter os valores apresentados na tabela a seguir;
- Comparou-se os dados obtidos pelo protótipo e os dados obtidos pelos multímetros e calculou-se a concordância entre as medições.

Abaixo, segue a tabela com os resultados obtidos:

Tabela 2 - Tabela comparativa entre as medições realizadas pelo protótipo e os multímetros.

	Tensão RMS (V)	Tensão RMS (multímetro) (V)	Concordância	Corrente RMS (I)	Corrente RMS (multímetro) (I)	Concordância
Lâmpada incandescente	128,6037104 ± 0,12	128,6 ± 1,028	99,99%	0,188314465 ± 0,00037	0,18 ± 0,0054	95,38%
Ferro de passar roupa	123,1389843 ± 0,12	122,95 ± 0,98	99,85%	8,004574713 ± 0,16	8,5 ± 0,255	94,17%
Bateadeira	127,6783627 ± 0,12	127,75 ± 1,022	99,94%	0,966694903 ± 0,0019	0,975 ± 0,029	99,15%

Fonte: Próprio autor

A nível de protótipo, foi possível alcançar resultados interessantes, obtendo uma concordância acima de 90%, provando que o firmware desenvolvido funciona corretamente na medição de grandezas obtidas pelo CI CS5463. Para trabalhos futuros, vale expandir os testes realizados.

5 CONCLUSÃO

Os objetivos elencados no início do trabalho foram alcançados. Foi muito construtivo pesquisar os itens da revisão bibliográfica, pois permitiu entender melhor os conceitos pesquisados. Além disso, foi interessante observar a importância dessas novas tecnologias para os diversos atores existentes.

O protótipo alcançou os resultados esperados. A nível de projeto, foi construído o medidor e este conseguiu tanto se comunicar via internet a partir do protocolo MQTT, quanto gerar dados relativamente confiáveis, graças ao CI CS5463. Foi possível desenvolver um protótipo de baixo custo, que pode ser melhorado em diversos pontos, otimizando custos e aumentando as funcionalidades existentes.

O protótipo possibilitou a participação na competição de sistemas embarcados da Intel, que por consequência possibilitou viajar para João Pessoa e apresentar o projeto para diversas pessoas. Além disso, foi escrito um artigo sobre o assunto, que está em fase de revisão.

Foi possível aprender muitas coisas, que normalmente não são aprofundadas na graduação, como, por exemplo: comunicação usando protocolos SPI e MQTT, estudo sobre métodos de medição de energia, aprofundamento dos conhecimentos na área de sistemas embarcados, uso de GIT, SSH e Linux em uma aplicação prática.

Por fim, o projeto ainda pode ser muito melhorado podendo inclusive virar um produto. Além disso, com a introdução de técnicas de IA nos dados obtidos pelo medidor, é possível criar novas e inovadoras funcionalidades.

REFERÊNCIAS

- [1] RIVERA, Ricardo; ESPOSITO, Alexandre Siciliano; TEIXEIRA, Ingrid. Redes elétricas inteligentes (smart grid): oportunidade para adensamento produtivo e tecnológico local. p43-84. Revista do BNDES, nº40, 2013.
- [2] AMIN, S. M.; WOLLENBERG, B. F. Toward a smart grid. IEEE Power and Energy Magazine, v. 3, n. 5, p. 34-38, 2005.
- [3] ABOBOREIRA, Felipe Luz; CRUZ, Antônia Ferreira dos Santos. A importância do Smart Grid na rede elétrica de distribuição do Brasil
- [4] VIEIRA, Daniel; SHAYANI, Rafael Amaral; OLIVEIRA, Marcos Aurelio Gonçalves. Net Metering in Brazil: regulation, opportunities and challenges. IEEE Latin America Transactions, v. 14, nº 8, p3687-3694, 2016.
- [5] Smart Grid Communications. NIST. Disponível em: <https://www.nist.gov/programs-projects/smart-grid-communications-0>. Acesso: 21 jan. 2021.
- [6] Redes elétricas inteligentes: contexto nacional. – Brasília, DF :Centro de Gestão e Estudos Estratégicos, 2012.
- [7] Agência Brasileira de Desenvolvimento Industrial – ABDI. Mapeamento da cadeia fornecedora de TIC e de seus produtos e serviços para redes elétricas inteligentes (REI). Relatório Consolidado. v.1, p412, 2017
- [8] SILVA, Marcelo Rubia da. Modelação e análise da vida útil (metrológica) de medidores tipo indução de energia elétrica ativa. 2010. 165 p. Dissertação (Mestrado). Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. São Paulo, 2010
- [9] ANEEL. Nota Técnica nº 0044/2010-SRD/ANEEL. 2012
- [10] EEI, AEIC, UTC. Smart Meters and Smart Meter Systems: A Metering Industry Perspective; p35, 2011.
- [11] Gartner Glossary. Gartner. Disponível em: <https://www.gartner.com/en/information-technology/glossary/advanced-metering-infrastructure-ami>. Acesso: 21 jan. 2021.
- [12] MOHASSEI, Ramyar Rashed; MOHAMMADI, Farah; FUNG, Alan S; RAAHEMIFAR, Kaamran. A survey on Advanced Metering Infrastructure. International Journal of Electrical Power & Energy Systems. 2014
- [13] Smart Grid. GTA UFRJ. 2021. Disponível em: https://www.gta.ufrj.br/ensino/eel878/redes1-2016-1/16_1/smartgrid/ . Acesso: 21 jan. 2021.
- [14] Tarifa Branca. Aneel. 2021. Disponível em: <https://www.aneel.gov.br/tarifa-branca>. Acesso: 21 jan. 2021.
- [15] Wiki da biblioteca UPM. Disponível em: <https://iotdk.intel.com/docs/master/upm/modules.html>. Acesso: 21 jan. 2021.
- [16] Wiki da biblioteca MRAA. Disponível em: <https://iotdk.intel.com/docs/master/mraa/python/index.html>. Acesso: 21 jan. 2021.

- [17] Protocolo MQTT. IBM. Disponível em: <https://developer.ibm.com/br/technologies/iot/articles/iot-mqtt-why-good-for-iot/>. Acesso: 21 jan. 2021.
- [18] MQTT – Protocolo para IoT. Embarcados. Disponível em: <https://www.embarcados.com.br/mqtt-protocolos-para-iot/>. Acesso: 21 jan. 2021.
- [19] Comunicação SPI – Parte 1. SACCO, Francesco. Embarcados. Disponível em: <https://www.embarcados.com.br/spi-parte-1/>. Acesso: 21 jan. 2021.
- [20] Datasheet do CI CS5463. Cirrus Logic. Disponível em: https://www.cirrus.com/jp/pubs/proDatasheet/CS5463_F3.pdf. Acesso: 21 jan. 2021.
- [21] AN366. Cirrus Logic. Disponível em: <http://www.cirrus.com/en/pubs/appNote/AN366REV2.pdf>. Acesso: 21 jan. 2021.
- [22] CRD5463PM. Cirrus Logic. Disponível em: http://www.cirrus.com/en/pubs/rdDatasheet/CRD5463PM_RD2.pdf. Acesso: 21 jan. 2021.
- [23] Edison Getting Started Guide. Sparkfun. Disponível em: https://learn.sparkfun.com/tutorials/edison-getting-started-guide?utm_source=Edison&utm_medium=button&utm_campaign=ProdPg&_ga=2.211699733.1886315701.1609972631-894189947.1609972631. Acesso: 21 jan. 2021.
- [24] Template de um dashboard. Klorofil. Disponível em: <https://www.themeinneed.com/downloads/klorofil-free-bootstrap-dashboard-template/>. Acesso: 21 jan. 2021.
- [25] API para gráficos em Javascript. Highcharts. Disponível em: <http://www.highcharts.com/products/highcharts>. Acesso: 21 jan. 2021.
- [26] Frases de Thomas Alva Edison. Disponível em: <https://www.frasesfamosas.com.br/frases-de/thomas-alva-edison/#:~:text=%E2%80%9CEu%20n%C3%A3o%20falhei%2C%20encontrei%2010,solu%C3%A7%C3%B5es%20que%20n%C3%A3o%20davam%20certo.%E2%80%9D&text=%E2%80%9CNossa%20maior%20fraqueza%20%C3%A9%20a,tentar%20apenas%20uma%20vez%20mais.%E2%80%9D>. Acesso: 21 jan. 2021.
- [27] Smart grid Evolution. Eolas. Disponível em: <https://www.eolasmagazine.ie/smart-grid-evolution/>. Acesso: 24 jan.2021.
- [28] Development and implementation of advanced metering infrastructure for efficient energy utilization in smart grid environment. MARIMUTHU, Krishna Paramathma; DURAIRAJ, Devaraj; SRINIVASAN, Swathikumar Karthik. Disponível em: <https://onlinelibrary.wiley.com/cms/asset/674d2203-08db-4055-9cf4-9d4026ea622b/etep2504-fig-0002-m.jpg>. Acesso: 24 jan. 2021
- [29] Tarifa branca. ANEEL. Disponível em: https://www.aneel.gov.br/sala-de-imprensa-exibicao-2/-/asset_publisher/zXQREz8EVIZ6/content/aneel-aprova-tarifa-branca-nova-opcao-para-os-consumidores-a-partir-de-2018/656877. Acesso em: 24 jan. 2021
- [30] Estrutura padrão publish-subscriber do MQTT, Wikipedia. BRUNO, Siol. Disponível em: https://pt.wikipedia.org/wiki/MQTT#/media/Ficheiro:Estrutura_padr%C3%A3o_publish-subscriber.png. Acesso em: 24 jan. 2021.

- [31] Nuevo Tutorial sobre el bus SPI. PROMETEC. Disponível em: <https://www.prometec.net/nuevo-tutorial-sobre-el-bus-spi/>. Acesso em: 24 jan. 2021.
- [32] Imagem Intel Edison. MAKER PRO. Disponível em: <https://maker.pro/storage/4H4tan4/4H4tan4bbOI3qb5EKGMbp4RzII PxEnsdsgB1IXag.jpeg>. Acesso em: 24 jan. 2021.
- [33] Imagem do sensor de corrente SCT013-100. FILIPEFLOP. Disponível em: https://uploads.filipeflop.com/2017/07/IMG_2909.png. Acesso em: 24 jan. 2021.
- [34] Imagem - Grove – LCD RGBBacklight. AppMarkers. Disponível em: <https://appmarkers.tistory.com/entry/Grove-LCD-RGB-Backlight>. Acesso em: 24 jan. 2021.
- [35] Smart grid distribution and smart meters. ST. Disponível em: https://www.st.com/content/ccc/resource/sales_and_marketing/promotional_material/brochure/eb/29/0b/3e/a3/7a/4b/7d/brmeter.pdf/files/brmeter.pdf/jcr:content/translations/en.brmeter.pdf . Acesso: 24 jan. 2021.

APÊNDICE A – PRODUÇÃO TÉCNICO-CIENTÍFICA

No que toca a produção técnico-científica, merece o destaque a duas atividades desenvolvidas – a participação nas finais do SBESC e a escrita de um artigo.

Falando um pouco sobre a competição, esta foi dividida em 3 fases, onde as duas primeiras se basearam na entrega de relatórios, visando mostrar qual é o projeto e o seu desenvolvimento durante o período de competição e a última fase, presencial, visou mostrar o que foi desenvolvido durante o período da competição e fazer as avaliações cabíveis. O protótipo conseguiu chegar até a última fase da competição, que foi realizada na cidade de João Pessoa durante o SBESC.

A competição e a presença no SBESC foi algo muito interessante e gratificante, pois, além da experiência de defender um projeto (com sucessivas apresentações e exposições) foi possível conhecer estudantes e professores de universidades de todo o país.

O artigo atualmente está em fase de revisão.