



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO  
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA  
MESTRADO EM ENGENHARIA ELÉTRICA

ANDERSON RODRIGUES DE MELO

**DESENVOLVIMENTO DE UM SISTEMA SUPERVISÓRIO APLICADO  
EM UMA USINA DE GERAÇÃO DE ENERGIA SOLAR  
FOTOVOLTAICA**

MOSSORÓ – RN

2020

ANDERSON RODRIGUES DE MELO

**DESENVOLVIMENTO DE UM SISTEMA SUPERVISÓRIO APLICADO  
EM UMA USINA DE GERAÇÃO DE ENERGIA SOLAR  
FOTOVOLTAICA**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Elétrica.

Linha de Pesquisa: Sistemas Elétricos

Orientadora: Profa. Dra. Fabiana Karla de Oliveira Martins Varella Guerra – UFERSA.

Coorientador: Prof. Dr. Marcelo Roberto Bastos Guerra Vale – UFERSA.

MOSSORÓ – RN

2020

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei n° 9.279/1996 e Direitos Autorais: Lei n° 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

M528d Melo, Anderson Rodrigues de.  
Desenvolvimento de um sistema supervisorio  
aplicado em uma usina de geração de energia solar  
fotovoltaica / Anderson Rodrigues de Melo. - 2020.  
80 f. : il.

Orientadora: Fabiana Karla de Oliveira Martins  
Varella Guerra.

Coorientador: Marcelo Roberto Bastos Guerra  
Vale.

Dissertação (Mestrado) - Universidade Federal  
Rural do Semi-árido, Programa de Pós-graduação em  
Engenharia Elétrica, 2020.

1. Geração Fotovoltaica. 2. IoT. 3. Medição de  
energia elétrica. 4. Sistema Supervisorio. 5.  
Blynk®. . I. Guerra, Fabiana Karla de Oliveira  
Martins Varella, orient. II. Vale, Marcelo  
Roberto Bastos Guerra , co-orient. III. Título.

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

ANDERSON RODRIGUES DE MELO

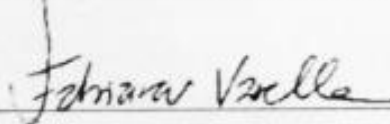
DESENVOLVIMENTO DE UM SISTEMA SUPERVISÓRIO APLICADO EM UMA  
USINA DE GERAÇÃO DE ENERGIA SOLAR FOTOVOLTAICA

Dissertação apresentada ao Mestrado em Engenharia Elétrica do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Mestre em Engenharia Elétrica.

Linha de Pesquisa: Sistemas Elétricos

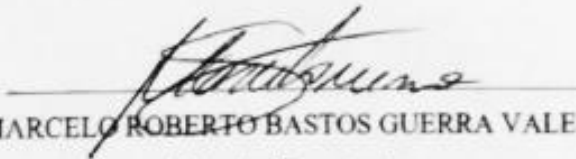
Defendido em: 24/07/2020

**BANCA EXAMINADORA**



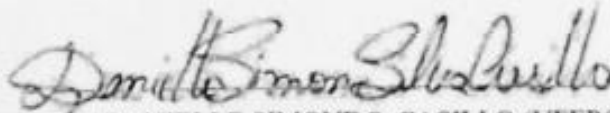
Profa. Dra FABIANA KARLA DE OLIVEIRA MARTINS VARELLA GUERRA (UFERSA)

Presidente



Prof. Dr MARCELO ROBERTO BASTOS GUERRA VALE (UFERSA)

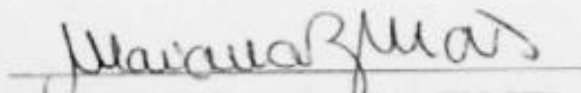
Membro Examinador



Profa. Dra DANIELLE SIMONE S. CASILLO (UFERSA)

Digitizado com CamScanner

Membro Examinador



Profa. Dra MAIANA BRITO DE MATOS (UFBA)

Membro Examinador

## **AGRADECIMENTOS**

Primeiramente a Deus que me proporcionou tamanha conquista e foi quem me ajudou em muitos momentos difíceis.

Aos meus pais Gilberto e Marta, por me incentivarem ao longo da caminhada e por serem um exemplo de determinação, coragem e amor. Por compreender minha ausência e apoio incondicional. E meu irmão Ederson, que me apoiou desde o início, fazendo das minhas conquistas as dele.

A minha orientadora, Prof.<sup>a</sup> Fabiana Varella, por todo o seu auxílio, sempre prestativa e contribuindo de forma significativa para elaboração deste trabalho, transmitindo importantes conhecimentos.

Ao meu coorientador, Prof. Marcelo Guerra, pelas importantes contribuições e por toda a disponibilidade a que se dispôs para me ajudar.

Aos meus amigos dentro e fora da UFERSA que tanto me ajudaram a ter forças para enfrentar os desafios de conduzir o mestrado em paralelo com o trabalho, auxiliando no que fosse possível.

Agradeço a minhas amigas Daisy, Ana e Aline, por acreditarem e dado o suporte necessário para a realização deste trabalho.

Enfim, agradeço a todos que contribuíram de alguma forma, para a realização deste sonho.

## RESUMO

O aumento da demanda por energia elétrica em virtude da melhoria da qualidade de vida, aliado à crescente preocupação com o meio ambiente, têm estimulado a procura por fontes alternativas de energia que reduzam os impactos ambientais. A energia solar fotovoltaica tem tido um papel de destaque nos últimos anos na geração de energia elétrica, e a mesma depende de diversos fatores, sendo o principal fator a irradiação solar incidente nos módulos fotovoltaicos. Mensurar a energia gerada por um sistema fotovoltaico é fundamental para que se possa identificar baixa eficiência ou falhas, portanto, mensurações preditivas permitem aumentar a eficiência do projeto, uma vez que o monitoramento contínuo otimiza a operação e identifica a necessidade de manutenção do sistema. Este trabalho tem como objetivo a aplicação de *Internet of Things* (IoT) para realizar a medição de parâmetros elétricos gerados por um inversor em operação na usina solar fotovoltaica da Universidade Federal Rural do Semi-Árido (UFERSA), visto que o mesmo não possui sistema supervisorio. Neste contexto, foi desenvolvido um dispositivo baseado em IoT para mensurar a energia gerada pela usina fotovoltaica, visando disponibilizar os dados em tempo real e histórico através de um sistema supervisorio. Foram realizados testes com o intuito de avaliar a acurácia e a interface de comunicação do protótipo, onde este se mostrou eficiente em ambos os quesitos.

**Palavras-chave:** Geração Fotovoltaica, IoT, Medição de energia elétrica, Sistema Supervisorio, *Blynk*®.

## ABSTRACT

The increase demand for electric energy due to the improvement in the quality of life has stimulated the demand for alternative renewable energy sources that, in addition to providing energy benefits, also reduce environmental impacts. Photovoltaic solar energy, for example, has played a prominent role in recent years, primarily in the generation of electricity. However, the use of this form of energy depends on several factors, the main one being the solar radiation on photovoltaic modules. Measuring the energy generated by a solar photovoltaic system is essential when it is desired to identify the efficiency rate or the existence of failures. Therefore, predictive measurements allow to increase the efficiency of the system, since the continuous monitoring optimizes the operation and identifies the need for its maintenance. Therefore, this work aims to apply Internet of Things (IoT) to perform the measurement of electrical parameters generated by an inverter operating in a solar photovoltaic plant at the Federal Rural University of the Semi-Arid (UFERSA), since it does not have a supervisory system, that is, it does not provide the desired monitoring. In this context, an IoT-based device was developed to measure the energy generated by the photovoltaic solar plant, to make data available in real time, as well as the history through a supervisory system. Tests were carried out in order to evaluate the accuracy and the communication interface of the prototype, where it proved to be efficient in both aspects.

**Key-words:** Photovoltaic generation. IoT. Electricity measurement. Supervisory System. Blynk®.

## LISTA DE FIGURAS

Figura 1: Comparação entre geração tradicional e geração distribuída.....	18
Figura 2: Conectividade e objetos inteligentes.....	24
Figura 3: Fluxograma com etapas para realização do trabalho de dissertação.....	30
Figura 4: Representação global do protótipo.....	31
Figura 5: Imagem física do TC utilizado no projeto (Modelo: SCT013-030).....	32
Figura 6: Divisor de tensão implementado.....	33
Figura 7: Esquemático de ligação do CI ADE7758. ....	33
Figura 8: Diagrama de blocos ADE7758. ....	34
Figura 9: Esquemático de ligação entre: Arduino Mega 2560, ESP-01 e ADE7758.....	35
Figura 10: Esquema elétrico da placa de medição .....	36
Figura 11: <i>Layout</i> físico da placa de medição .....	37
Figura 12: Diagrama elétrico e <i>layout</i> físico da placa de comunicação <i>Wi-Fi</i> .....	37
Figura 13: IDE Arduino.....	39
Figura 14: Componentes da plataforma <i>Blynk</i> . ....	40
Figura 15: Arquitetura utilizando <i>Raspberry Pi</i> como servidor local. ....	42
Figura 16: Iniciação do servidor local do <i>Blynk</i> ®. ....	42
Figura 17: Configurações do aplicativo <i>Blynk</i> ®, instalado no dispositivo móvel. ....	43
Figura 18: <i>Widget Box</i> e balanço de energia. ....	44
Figura 19: Informações do status no servidor local <i>Blynk</i> ®. ....	44
Figura 20: Valor lido pelo <i>firmware</i> com valor lido no analisador. ....	45
Figura 21: Regulador de tensão, banco de resistência e TC's. ....	46
Figura 22: Fluxograma de calibração do projeto.....	46
Figura 23: Calibração da tensão eficaz.....	47
Figura 24: Calibração da corrente eficaz.....	47
Figura 25: Processos realizados pelo projeto. ....	49
Figura 26: Fluxograma de teste de campo (calibração).....	50
Figura 27: Ligações para medição do protótipo e analisador de energia. ....	51
Figura 28: Leitura da tensão do protótipo e analisador de energia.....	51
Figura 29: Leitura da corrente do protótipo e analisador de energia.....	52
Figura 30: Leitura da potência do protótipo e analisador de energia. ....	52



Figura 31: Fluxograma de teste de campo (comunicação com supervisor). .....	53
Figura 32: Placa com ADE7758, Arduino Mega e ESP-01 (a) e <i>Raspberry Pi3</i> (b). .....	53
Figura 33: Interface com o usuário marcando valores instantâneo e acumulado. ....	54
Figura 34: Gráfico dos valores lidos de tensão, corrente e potência. ....	54
Figura 35: Valor medido pelo protótipo e faixa do erro ( $\pm 2,5$ ). ....	55

## LISTA DE TABELAS

Tabela 1: Unidades Consumidoras (UC's) com geração distribuída em junho/2020. ....	20
Tabela 2: Erro de leitura do <i>firmware</i> . ....	48
Tabela 3: Valor lido pelo Arduino (Tensão A) x valor lido pelo analisador. ....	74
Tabela 4: Valor lido pelo Arduino (Corrente A) x valor lido pelo analisador. ....	74

## LISTA DE ABREVIATURAS E SIGLAS

- ABSOLAR – Associação Brasileira de Energia Solar Fotovoltaica.
- AC – *Alternating Current* (Corrente alternada)
- AMI – *Advanced Metering Infrastructure* (Infraestrutura avançada de medição)
- AMM – *Advanced Metering Management* (Gerenciamento avançado de medição)
- AMR – *Automated Meter Reading* (Leitura automatizada do medidor)
- ANEEL – Agência Nacional de Energia
- CAPEX – *Capital Expenditure* (Despesa de Capital)
- CI – Circuito Integrado
- CGEE – Centro de Gestão e Estudos Estratégicos
- CITED – Centro Integrado de Inovação Tecnológica do Semiárido
- CoAP – *Constrained Application Protocol* (Protocolo de Aplicação Restrita)
- CRESESB – Centro de Referência para Energia Solar e Eólica Sérgio de Salva Brito
- DC – Direct Current (Corrente contínua)
- DEOE – *United States Department of Energy's Office of Electricity* (Escritório de Eletricidade do Departamento de Energia dos Estados Unidos)
- EPE – Empresa de Pesquisa Energética
- IEEE – *Institute of Electrical and Electronics Engineers* (Instituto de Engenheiros Elétricos e Eletrônicos)
- GD – Geração Distribuída
- IEA – *International Energy Agency* (Agência Internacional de Energia)
- IHM – Interface Homem Máquina
- INEE – Instituto Nacional de Eficiência Energética.
- INMETRO – Instituto Nacional de Metrologia, Qualidade e Tecnologia
- IoT – *Internet of Things* (Internet das Coisas)
- IP – *Internet Protocol* (Protocolo da Internet)
- MMGD – Microgeração e Minigeração Geração Distribuída
- MMI – Ministério de Minas e Energia
- MQTT – *Message Queue Telemetry Transport* (Transporte de Telemetria da Fila de Mensagens)
- PCI – Placa de Circuito Impresso
- PLC – *Power Line Communications* (Comunicação por Linha de Energia)
- PPGEE - Programa de Pós-Graduação em Engenharia Elétrica

REN21 – *Renewable Energy Policy Network for the 21st Century* (Rede de Políticas de Energia Renovável para o Século XXI)

RMS – *Root Mean Square* (Valor Quadrático Médio)

SENAI – Serviço Nacional de Aprendizagem Industrial

SIN – Sistema Interligado Nacional

SPI – *Serial Peripheral Interface* (Interface Periférica Serial)

TC – Transformador de Corrente

UC's – Unidades Consumidoras

UFERSA – Universidade e Federal Rural do Semi-Árido

UFV – Usina Fotovoltaica

## LISTA DE SÍMBOLOS

A – Ampère

GW - Gigawatt

kA – Quiloampère

kV - Quilovolt

kW – Quilowatt

kWp – Quilo-watt pico

MW – Megawatt

TWh – Terrawatt-hora

V – Volt

W – Watt

$\varepsilon$  – Erro

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO .....</b>	<b>15</b>
1.1	Objetivo geral .....	16
1.2	Objetivos específicos .....	17
1.3	Organização do trabalho .....	17
<b>2.</b>	<b>REFERENCIAL TEÓRICO .....</b>	<b>18</b>
2.1	Geração Distribuída.....	18
2.1.1	Geração solar fotovoltaica.....	20
2.2	<i>Smart Grid</i> .....	22
2.3	<i>Internet of Things (IoT)</i> .....	23
2.3.1	Protocolos MQTT .....	25
2.3.2	Sistema de supervisão (Supervisório).....	25
2.4	Trabalhos desenvolvidos na área.....	27
<b>3.</b>	<b>MATERIAIS E MÉTODOS .....</b>	<b>29</b>
3.1	Local da pesquisa .....	29
3.2	Procedimento da pesquisa .....	30
3.3	Desenvolvimento do projeto.....	31
3.3.1	<i>Hardwares utilizados</i> .....	32
3.3.1.1	Transformador de Corrente (TC) .....	32
3.3.1.2	Divisor de tensão.....	33
3.3.1.3	Circuito integrado ADE7758 .....	34
3.3.1.4	Arduino Mega 2560 .....	35
3.3.1.5	ESP-01 (módulo <i>Wi-Fi</i> ) .....	35
3.3.1.6	Placas de circuito impresso (PCI) .....	36
3.3.1.7	<i>Raspberry Pi</i> .....	37
3.3.1.8	Roteador <i>Wi-Fi</i> .....	38
3.3.2	<i>Softwares utilizados</i> .....	38
3.3.2.1	Implementação do <i>firmware</i> .....	38
3.3.2.2	Plataforma <i>Blynk</i> ® .....	39
3.3.3	<b>Processo do projeto .....</b>	<b>40</b>
3.3.3.1	Leitura das grandezas elétricas .....	41
3.3.3.2	Processamento dos valores lidos e envio para o servidor .....	41
3.3.3.3	Disponibilização dos valores lidos e armazenamento .....	42
3.3.3.4	Visualização dos dados .....	43

3.3.4	Calibração do projeto .....	45
4.	RESULTADOS E DISCUSSÕES .....	49
4.1	Testes e validação .....	50
4.1.1	Teste de acurácia na usina fotovoltaica do CITED (calibração) .....	50
4.1.2	Teste de comunicação do protótipo (monitorando Usina Fotovoltaica) .....	53
4.2	Avaliação dos testes.....	55
5.	CONCLUSÃO.....	57
	REFERÊNCIAS .....	59
	APÊNDICE A .....	64
	APÊNDICE B.....	72

## 1. INTRODUÇÃO

O aumento da demanda por energia e a busca por fontes de geração consideradas limpas e sustentáveis, fazem com que a implementação de projetos de energia solar tenha um papel fundamental no cenário energético.

No cenário mundial a geração de energia solar fotovoltaica teve um aumento de 22% em 2019, que representa um aumento de geração estimado de 131 Terawatt-hora (TWh), sendo o segundo maior crescimento absoluto de geração de energia renovável, ficando um pouco atrás da energia eólica e a frente da hidráulica. Representando aproximadamente 3% da matriz energética de geração de eletricidade, sendo a terceira maior tecnologia de eletricidade renovável (IEA, 2020).

Na última década os avanços tecnológicos têm ocorrido em períodos cada vez mais curtos e nas mais diversas áreas, sendo uma delas a energia elétrica, em que a rede elétrica tradicional composta por linhas de transmissão, subestações, transformadores e rede de distribuição fornece eletricidade da usina para o consumidor, avança para uma rede que possa automatizar e gerenciar a crescente demanda de forma mais eficiente, com participação conjunta da concessionária junto com o consumidor.

O Brasil tem caminhado para o aprimoramento da interação entre o consumidor e a concessionária. Em Agência Nacional de Energia Elétrica (ANEEL) (2015), trata da Resolução Normativa nº 687/2015 que atualiza a Resolução Normativa nº 482/2012 esclarecendo o modelo de compensação de geração distribuída e em ANEEL (2016), trata da Resolução Normativa nº 733/2016 que possibilita aplicação de modalidade tarifária horária branca para consumidor de baixa tensão, tornando necessário a implementação de medidores eletrônicos, que são fundamentais para possibilitar a medição bidirecional no caso de geração distribuída e consumo horário, para modalidade de tarifária horária branca.

Atualmente a capacidade instalada de geração por Usinas Fotovoltaicas (UFV) no Brasil representa cerca de 1,68% da matriz energética, cujos empreendimentos em construção iniciadas e não iniciadas representam, respectivamente, cerca de 5,68% e 43,50% do total de potência outorgada, demonstrando a tendência de aumento de aplicação da energia fotovoltaica (ANEEL.a, 2020).

Os sistemas fotovoltaicos representam cerca de 99,8% dos sistemas de geração distribuída (ABSOLAR, 2020), sendo desejável a instalação de monitoramento de sistemas fotovoltaicos para auxiliar na detecção de falhas ou perda de desempenho, de modo que possa elevar a eficiência global do sistema (CRESESB, 2014).



A tecnologia solar fotovoltaica será a segunda fonte de maior incentivo para atendimento à demanda no plano decenal (2019-2029), que prevê um acréscimo de 7.000 MW de indicativa adicional, podendo chegar até 15.000 MW se manter a tendência de queda do custo de investimento (*Capital Expenditure* - CAPEX), não contabilizando a perspectiva de aumento da geração distribuída, onde se estima que 85% utiliza a tecnologia fotovoltaica (EPE, 2020).

Segundo ABSOLAR (2020), a geração distribuída no Brasil obteve um aumento de 77,83% no primeiro trimestre de 2020 (em comparação ao mesmo período de 2019) e possui tendência de crescimento do aumento da geração distribuída fotovoltaica no segundo semestre. Demonstrando assim que o consumidor brasileiro identificou uma forma de redução de custos, reduzindo a dependência das bandeiras tarifárias amarela ou vermelha e aumentos das distribuidoras. O monitoramento da energia gerada por sistemas fotovoltaicos, seja em grandes usinas fotovoltaicas ou pequenas unidades consumidoras, é de fundamental importância para a rentabilidade e eficiência do projeto, visto que a rápida detecção de uma baixa eficiência ou falhas ocasionam um menor tempo de perda de geração.

### **1.1 Objetivo geral**

O presente trabalho tem como objetivo geral projetar e desenvolver um dispositivo para monitoramento, aquisição e armazenamento de parâmetros elétricos de uma usina solar fotovoltaica na Universidade Federal Rural do Semi-Árido, Mossoró/RN, com o intuito de se acompanhar os parâmetros em tempo real, bem como o histórico da mesma.

## 1.2 Objetivos específicos

Com o intuito de alcançar o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Projetar e desenvolver um *hardware* capaz de realizar a leitura das grandezas elétricas geradas por um sistema fotovoltaico, utilizando o ADE7758<sup>1</sup>;
- Utilizar o conceito de IoT para tratamento e transmissão dos dados coletados através da placa de prototipagem Arduino Mega 2560 em conjunto com ESP-01;
- Implementar sistema de servidor local e *dashboard* (tela de monitoramento) que possibilite o usuário observar a geração em tempo real e histórico, através de uma rede local dedicada;
- Realizar teste de acurácia para avaliar se os dados medidos apresentam erro inferior a 2,5%;
- Realizar testes de comunicação para avaliar se os dados medidos estão disponíveis no aplicativo e armazenados no servidor remoto.

## 1.3 Organização do trabalho

No capítulo 2, será apresentado o referencial teórico descrevendo a respeito de geração distribuída, geração de energia fotovoltaica, descrição a respeito de *Smart Grid*, descrição a respeito de internet das coisas, sistema supervisorio e por último os trabalhos desenvolvidos na área.

No capítulo 3, são apresentados o local de desenvolvimento da pesquisa, etapas que foram seguidas para obtenção do projeto, desenvolvimento e calibração do protótipo desenvolvido.

No capítulo 4, são analisados e discutidos os resultados obtidos nos testes de validação do protótipo, assim como detalhado dificuldades tidas durante o desenvolvimento do mesmo.

No capítulo 5 serão apresentadas as conclusões e sugestões de trabalhos futuros com a mesma temática do trabalho realizado.

---

<sup>1</sup> Circuito integrado de alta precisão, utilizado para medição de energia elétrica trifásica (ANALOG DEVICE©, 2019).

## 2. REFERENCIAL TEÓRICO

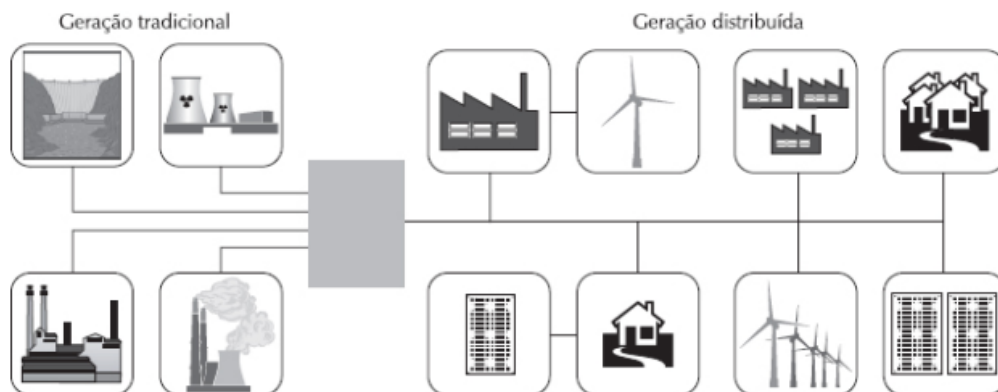
Neste Capítulo serão abordados assuntos relevantes para a caracterização do trabalho, sendo finalizado com os trabalhos já desenvolvidos na área e que obtiveram êxito na sua implementação. Será descrito a respeito da geração distribuída de energia elétrica, o cenário atual das fontes de energia utilizadas dando ênfase na geração fotovoltaica. Será destaque a respeito de redes inteligentes (*Smart Grid*) que tem como base os medidores eletrônicos, descrição a respeito de internet das coisas, sistema supervisorio e trabalhos desenvolvidos na área.

### 2.1 Geração Distribuída

A Geração distribuída (GD) é a denominação que se dá quando a geração de energia elétrica é realizada próxima aos locais de consumo. Tal geração chegou a ser regra na primeira metade do século, quando a energia era gerada localmente, no entanto, a partir da década de 40 quando a geração em usinas de grande porte foi barateada, foi reduzido o interesse pela GD, bem como o incentivo para desenvolvimento deste tipo de tecnologia (INEE, 2020).

O modelo de geração distribuída se opõe ao modelo tradicional, onde se utilizam grandes usinas que se localizam distantes dos locais de consumo, e tem as fontes alternativas de energia como o maior potencial, se destacando as energias fotovoltaica e eólica (que apresentam maior potencial para geração). Pode ser verificado na Figura 1, a representação de geração distribuída (VILLALVA, 2015).

Figura 1: Comparação entre geração tradicional e geração distribuída



Fonte: VILLALVA, 2015.

No Brasil a geração distribuída foi definida oficialmente com o Decreto nº 5.163 de 30 de julho de 2004:

Art. 14. [...], considera-se geração distribuída a produção de energia elétrica proveniente de empreendimentos de agentes concessionários, permissionários ou autorizados, incluindo aqueles tratados pelo art. 8º da Lei nº 9.074, de 1995, conectados diretamente no sistema elétrico de distribuição do comprador, exceto aquela proveniente de empreendimento:

I - hidrelétrico com capacidade instalada superior a 30 MW; e  
II - termelétrico, inclusive de cogeração, com eficiência energética inferior a setenta e cinco por cento, conforme regulação da ANEEL, a ser estabelecida até dezembro de 2004.

*Parágrafo único.* Os empreendimentos termelétricos que utilizem biomassa ou resíduos de processo como combustível não estarão limitados ao percentual de eficiência energética prevista no inciso II do caput (BRASIL, 2004).

Em 17 de abril de 2012, quando entrou em vigor a Resolução Normativa ANEEL nº482/2012, foi possível que o consumidor passasse a gerar sua própria energia elétrica através de fontes renováveis ou cogeração qualificada e inclusive fornecer o excedente para a rede de distribuição que está conectada. A Resolução Normativa 687/2015 foi elaborada com o intuito de estimular a implementação de projetos, tendo como exemplos de mudanças o aumentando o público alvo (permitindo o uso de qualquer fonte renovável, além da cogeração qualificada), aumento do prazo de validade dos créditos, possibilidade de instalação de geração distribuída em condomínios, entre outras medidas de incentivo (ANEEL, 2015).

A geração distribuída é dividida em microgeração distribuída (potência instalada até 75 kW) e minigeração (com potência instalada superior a 75kW e menor ou igual a 5 MW), onde a energia ativa injetada pela unidade consumidora é cedida à distribuidora local e posteriormente compensada no consumo de energia ativa, incentivando o desenvolvimento de projetos principalmente da área de energia solar fotovoltaica (ANEEL, 2015).

A modalidade de Microgeração e Minigeração Geração Distribuída (MMGD) no Brasil, em junho de 2019 alcançou o marco de 1 Gigawatt (GW), superando as expectativas que o setor tinha para esta modalidade, cuja capacidade instalada em 2018 foi de 400 MW, o que representa a mesma capacidade adicionada em térmicas a gás natural no mesmo ano. A disponibilidade de potencial renovável, o alto valor das tarifas e o modelo de compensação de créditos extremamente favorável, foram os principais motivos da expansão (EPE, 2020).

De acordo com ABSOLAR (2020), entre as tecnologias de MMDG implementadas no Brasil, tem-se como destaque principal a geração fotovoltaica, que apresenta maior potencial devido sua modularidade, tendência de queda de custo, melhora na eficiência e difusão da tecnologia entre a sociedade, atualmente já representa 99,8% das conexões de MMDG, sendo mostrado na Tabela 1.

Tabela 1: Unidades Consumidoras (UC's) com geração distribuída em junho/2020.

<b>Tipo</b>	<b>Quantidade</b>	<b>Quantidade de UC's que recebem os créditos</b>	<b>Potência instalada (kW)</b>	<b>Percentual potência instalada (%)</b>
CGH	107	8.479	102.805,71	3,3%
EOL	64	108	10.413,86	0,3%
UFV	249.730	312.456	2.931.646,67	94,1%
UTE	240	4.598	71.856,54	2,3%

Fonte: Modificado de ANEEL.b, 2020.

De acordo com a Tabela 1, pode-se verificar que a potência instalada de geração fotovoltaica representa 94,1% da potência total de unidades consumidoras com geração distribuída, sendo está detalhada na subseção 2.1.1.

### **2.1.1 Geração solar fotovoltaica**

Os sistemas fotovoltaicos convertem diretamente a energia do sol para energia elétrica através do efeito fotovoltaico, que capta diretamente a luz solar, produzindo corrente elétrica. Esta corrente elétrica é processada por dispositivos controladores e conversores, podendo ser armazenada ou utilizada diretamente em sistemas elétricos (VILLALVA, 2015).

De acordo com Villalva (2015), os sistemas de geração de energia fotovoltaica são divididos em duas categorias: sistemas autônomos e sistemas conectados à rede. Sistemas autônomos ou sistemas isolados, são empregados em locais não atendidos por uma rede elétrica (exemplo: residências em zonas rurais, *camping*, iluminação pública, outros). Os sistemas conectados à rede têm o objetivo de gerar eletricidade para consumo local, podendo reduzir o consumo ou mesmo gerar excedente de energia para a rede elétrica.

Segundo Villalva (2015) e REN21 (2020), a energia fotovoltaica é uma das fontes de energia que mais cresce em todo o mundo. Em âmbito mundial a capacidade instalada de geração de energia fotovoltaica cresceu cerca de 12% em 2019, representando uma adição de aproximadamente 115 GW, sendo a tecnologia líder em implementação, representando cerca

de 57% das adições de potência instalada de energia renovável. A América Latina, segue a tendência de expansão, sendo favorecida devido à grande abundância de recursos solares, queda nos preços e climas políticos favoráveis, tendo como principais países o Brasil (2 GW), México (com quase 2 GW) e Argentina (0,5GW) (REN21, 2020).

A geração de energia solar fotovoltaica é uma tecnologia descentralizada, modular, silenciosa, limpa e renovável. No passado, esteve voltada apenas para carregamento de baterias em regiões sem acesso à energia elétrica e atualmente é a fonte dominante na produção de energia elétrica no setor residencial no Brasil (CGEE, 2012), (ANEEL.a, 2020).

Com o aumento do número de instalações de sistemas fotovoltaicos após as regulamentações da geração distribuída, o Comitê Brasileiro de Eletricidade (COBEI), foi responsável pela elaboração da Norma Brasileira (NBR) 16149, que trata das características de interface de conexão com a rede elétrica de distribuição de sistemas fotovoltaicos e a NBR 16150 trata dos ensaios de conformidade dos sistemas fotovoltaicos (CRESESB, 2014).

Ainda segundo CRESESB (2014), um sistema fotovoltaico utilizado em geração distribuída é constituído por um bloco gerador (que é constituído principalmente pelos módulos fotovoltaicos e acessórios), por um bloco de condicionamento de potência (inversor de frequência, dispositivos de proteção, supervisão e controle, e outros) e eventualmente bloco de armazenamento que são utilizados em operação ilhada do sistema de geração no caso de falta de energia elétrica (constituído por baterias).

O sistema fotovoltaico é bastante confiável, porém devido, geralmente, ter um difícil acesso e ser silencioso, ocasiona uma maior dificuldade na detecção de falhas, prejudicando o desempenho global do sistema, além do prejuízo financeiro pela ausência de geração (CRESESB, 2014).

Para contornar este problema em sistemas de maior porte e complexidade é implementado um sistema de supervisão e controle para otimizar a operação, onde os valores dos parâmetros elétricos são importantes para avaliação, tendo como destaque a quantidade de energia gerada. Na subseção 2.2, será descrito como é realizado a medição de energia elétrica (CRESESB, 2014).

## 2.2 *Smart Grid*

O termo *Smart Grid* surgiu pela primeira vez em 2005, presente no artigo publicado por Amin e Wollenberg (2005), com o título de “*Toward a Smart Grid*”, o qual tem como objeto o aumento da segurança, agilidade e robustez de uma infraestrutura de energia elétrica de enfrentar ameaças e condições imprevistas.

Segundo Hassan (2018) e IEEE (2020), *Smart Grid* é uma estrutura avançada de comunicação bidirecional de informações, controle de equipamentos e distribuição de energia elétrica, incluindo medidores inteligentes, aparelhos inteligentes e recursos de energia renovável.

De acordo com DEOE (2020), *Smart Grid* não trata apenas de utilitários e tecnologia, mas sim de fornecer informações e ferramentas para que o usuário possa fazer escolhas sobre o uso, semelhante ocorre com cartão de crédito, em que pode se acompanhar o consumo durante o mês corrente, o usuário poderá acompanhar o consumo de eletricidade do mês, sem necessitar aguardar o extrato mensal chegar, por meio de medidores inteligentes.

De acordo com Kanter e Litow (2009), as cidades inteligentes são aquelas capazes de conectar informações de sua infraestrutura física para melhorar as conveniências, aumentar as facilidades e identificar e solucionar problemas rapidamente.

As primeiras implementações de sistemas com alguma inteligência na rede foram possibilitadas através da medição eletrônica, tendo como exemplo no início de 2000, foi desenvolvido um projeto na Itália, instalando medição eletrônica em larga escala de cerca de 30 milhões de medidores eletrônicos com capacidade de comunicação via *Power Line Communications* (PLC), possibilitando a comunicação remota entre a concessionária e com o medidor (BOTTE; CANNATELLI; ROGAI, 2005).

Segundo Camargo (2009), algumas vantagens da implantação do sistema de redes inteligentes referem-se ao intercâmbio de informações, tornando as operações ágeis, reduzindo desperdícios, identificando rapidamente falhas e tendo uma melhor eficiência no sistema.

Em Rigodanzo (2015), a implementação do *Smart Grid* não se limita na medição eletrônica – *Automated Meter Reading* (AMR), passando pela eletrônica bidirecional com adição de algum tipo de controle – *Advanced Metering Infrastructure* (AMI) e evoluindo para gestão avançada da medição – *Advanced Metering Management* (AMM).

De acordo com Ye, Qian e Hu (2018), entre as várias vantagens na aplicação de *Smart Grid* se destacam uma melhor adaptação para recursos de geração de energia renovável,

aumento da eficiência das operações da rede e redução de perdas e por fim uma melhor confiabilidade e segurança no sistema.

Cada país toma a iniciativa de implementação com motivadores diferentes, onde de modo geral a Europa investe como ação para redução das emissões, Estados Unidos com o foco na melhoria da eficácia da rede e Ásia/ Pacífico como ferramenta que possibilite atender a demanda e busca por mercado tecnológico emergente. No Brasil, a implementação de *Smart Grid*, atenderia na melhoria da qualidade do serviço de energia e redução das perdas (CGEE, 2012).

Há uma perspectiva do mercado de energia dos países da América do Sul (liderados pelo Brasil), que haja um investimento de US \$ 18.1 bilhões em redes inteligentes na próxima década, sendo que os investimentos dependem da recuperação global da pandemia do COVID-19 e suas consequências econômicas (SMART ENERGY INTERNATIONAL, 2020).

### **2.3 Internet of Things (IoT)**

O advento da Indústria 4.0, que integra o sistema produtivo, computador e dispositivos móveis interligados à internet ou à intranet. Tem como objetivo de programar, gerenciar, controlar e cooperar em qualquer lugar do globo que tenha acesso a está rede com foco em otimizar e agregar valor, se torna vital a mudança de paradigma de como se utiliza a internet, internet das coisas (IoT), sendo diferencial para aumentar a competitividade das empresas e reduzir custos operacionais (SACOMANO; GONÇALVES; SILVA; BONILLA; SÁTYRO; MORAES; FERIGATTO; JÚNIOR; JACINTHO; REIS, 2018).

De acordo com Sacomano, Gonçalves, Silva, Bonilla, Sátyro, Moraes, Ferigatto, Júnior, Jacintho e Reis (2018), IoT, consiste em conectar objetos utilizados diariamente à internet, de forma a poderem ser acessados remotamente, através de dispositivos móveis como celular, notebook e outros, que tenham conexão à internet, podendo eles comunicarem entre si. Esses objetos podem ser dotados de capacidade de processamento, onde passam a ser chamados de “inteligentes” (*Smart*).

Conforme padrão IEEE Std 2413, a definição simples de internet das coisas é:

Um sistema de entidades (incluindo dispositivos *cyber*-físico, recursos de informações e pessoas) que trocam informações e interagem com o mundo físico, detectando, processando e atuando (IEEE, 2020).

Samie, Bauer e Henkel (2016) afirmam que a internet das coisas possui uma vasta gama de aplicações, no qual se destaca o grande potencial em aplicações de cuidados com a saúde

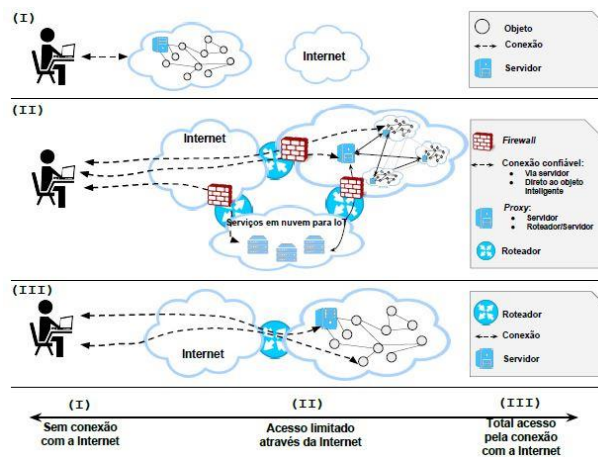


(monitoramento de dados vitais em tempo real, entre outros), assistência a vida (navegação em cidades, detecção de obstáculos, e etc.), prédios e casas inteligentes (maior conforto e redução de custos), cidades inteligentes (monitoramento de tráfego, temperatura, poluição, estacionamentos, etc.) e indústrias inteligentes (monitorar e agregar valor aos itens produzidos).

De acordo com Santos et al. (2016) e Samie Bauer e Henkel (2016), os dispositivos inteligentes aplicados em sistemas de IoT, apresentam em suas estruturas quatro unidades, em que uma é responsável pela aquisição de dados (sensores), uma pelo processamento dos dados (microcontrolador), uma pelo armazenamento dos dados (manter registro dos dados “log”) e a última responsável pela comunicação com a rede (com ou sem fio).

A comunicação com a rede pode ser realizada através de diferentes tecnologias, conectando dispositivos IoT em redes locais, assim como a internet. Como exemplo, tem-se o padrão Ethernet, presente em grande parte das redes locais com fio atualmente existentes e a tecnologia *Wi-Fi*, que faz parte do cotidiano da população, seja nas casas, escritórios, indústrias e até em espaço público. O padrão *ZigBee* possui características de baixa vazão de dados, reduzido consumo energético e baixo custo. Na Figura 2, são apresentados modelos de conectividade de objetos inteligentes (SANTOS et al., 2016).

Figura 2: Conectividade e objetos inteligentes.



Fonte: SANTOS et al., 2016.

Na Figura 2 são mostradas três configurações de objetos inteligentes, em que na primeira parte é mostrada a rede autônoma de objetos inteligentes, em que os objetos não possuem conexão com a Internet “pública”, sendo eles acessíveis por rede interna. Na segunda parte, a rede de objetos é limitada, pois o acesso aos dispositivos é restrito e por fim, todos objetos estão conectados à internet, “IoT autêntica” (SANTOS et al., 2016).

O modelo básico de arquitetura para objetos inteligentes apresenta três camadas, em que a primeira camada representa os objetos físicos onde os sensores coletam e processam as

informações, sendo ela conhecida como camada de percepção. A camada de rede é onde realiza o gerenciamento, roteamento e identificação das tecnologias de comunicação. Por fim, a camada de aplicação, a qual é responsável por prover serviços para os clientes (SANTOS et al., 2016).

Em virtude dos equipamentos IoT possuem poder computacional restrito, o protocolo de comunicação utilizado por eles tem como prerrogativa transmitir a informação com baixo poder computacional, destacando-se dois tipos de protocolo *Constrained Application Protocol* (CoAP) e o *Message Queue Telemetry Transport* (MQTT) (SANTOS et al., 2016).

### 2.3.1 Protocolos MQTT

O protocolo MQTT que foi criado pela IBM e subsequentemente repassado para o Grupo OASIS tem como estratégia de comunicação de *publish/subscribe*, em que os dispositivos se registram (*subscribe*) a um *broker* (servidor que controla a comunicação) para obter informações e avisar sempre que for publicado (*publishers*) informações através do *broker*, sendo assim uma comunicação assíncrona (SHINDE, 2016), (SANTOS et al., 2016).

De acordo com Gündoğran, Kietzmann, Lenders, Petersen, Schmidt e Wählich (2018) o MQTT é considerado um protocolo leve pois fornece uma estrutura de cabeçalho enxuta, que reduz a análise de pacotes e o torna adequado para dispositivos IoT com baixo custo de energia e sua facilidade de implementar.

### 2.3.2 Sistema de supervisão (Supervisório)

Mesmo existindo várias tecnologias de controle, é imprescindível que exista interface entre o controle e o monitoramento, proporcionando integração entre o homem e a máquina. Esta interface é denominada IHM (Interface Homem Máquina), podendo ser realizada por anunciadores de alarmes, sinaleiros, *displays*, teclados e botão de comandos ou através de um *software* supervisório (SENAI, 2013).

De acordo com Rosário (2005), sistema supervisório pode ser definido como uma interface amigável entre homem-máquina em que é utilizado tecnologia de computação e comunicação que permite que um sistema automatizado possa ser supervisionado e/ou controlado assumindo elevada relevância para o processo, obtendo como resultado redução de custos de desenvolvimento e manutenção.

Um sistema supervisório torna a comunicação do sistema monitorado com o mundo externo mais simples, gerenciando variáveis do processo, podendo ser guardadas em bancos de

dados locais ou remotos para fins de registro histórico. Eles podem ser concebidos com diversas configurações, sendo a de supervisão, comando e controle (SENAI, 2013).

Em geral um sistema supervisorio se propõem atender alguns requisitos como interface amigável com operador, geração automática de relatórios, facilidade para interação com outros aplicativos, acesso automático a banco de dados, acesso compartilhado e remoto, conexão em rede e gerenciamento de alarmes (ROSÁRIO, 2005).

O sistema de energia tornou-se cada vez mais vulnerável com adição de sistemas renováveis, tornando necessário o monitoramento e controle de modo remoto dos dispositivos conectados à rede principalmente quando a capacidade exceder os limites de estabilidade (SARINDA; IQBAL; MANN, 2017).

Em instalações fotovoltaicas (FV) o sistema de monitoramento pode ser um sistema avançado, simples ou somente com as informações fornecidas diretamente no inversor. O sistema de aquisição e armazenamento de dados pode permitir apenas a coleta dos dados no local ou possibilitar de forma remota através de dispositivos móveis ou computador que estejam conectados numa rede local ou internet (CRESESB, 2014).

De acordo com Portal Solas S.A. (2020), para atender uma demanda dos parceiros que buscavam solução simples para alertar sobre a iminência de panes no sistema de geração fotovoltaica, a equipe de engenharia da Elsys desenvolveu um dispositivo que permite o usuário acompanhar em qualquer lugar do mundo informações sobre seu consumo ou geração de energia elétrica, através de um dispositivo de medição conectado à nuvem via *Wi-Fi* e compatível com o protocolo MQTT.

A aplicação de medidor em sistemas fotovoltaicos permite o monitoramento do consumo e da energia gerada, assim é possível acompanhar em tempo real o saldo da energia gerada em relação à energia consumida, possibilitando o cliente comparar com os relatórios disponibilizados pela concessionária (PORTAL SOLAR S.A., 2020).

Em Sarinda, Iqbal e Mann (2017), inversores de baixo custo que podem ter o mesmo desempenho dos inversores de ponta, porém não possuem sistema de monitoramento embutido. Foram avaliados métodos e *softwares* para serem utilizados como sistema de monitoramento de baixo custo, onde a plataforma Blynk possui como característica a fácil instalação, fácil desenvolvimento e monitoramento apenas através do aplicativo em dispositivo móvel.

Segundo CRESESB (2014), a complexidade do sistema de supervisão instalado em projetos FV deve ser proporcional ao prejuízo que se terá devido a indisponibilidade de energia, devendo-se levar em conta a maneira que os dados serão transmitidos, podendo inviabilizar a instalação do mesmo, devido ao custo de sua implementação.

## 2.4 Trabalhos desenvolvidos na área

Vollmer, Bernhard, Schirrmeister, Ulrich e Valenzuela (2017), construíram um sistema para monitoramento das máquinas elétricas que consiste numa unidade de medição e controle, em que foi utilizado o ADE7758 para realizar medição dos parâmetros elétricos em conjunto com o Arduino Uno que se comunica ao Raspberry Pi através da porta USB. O Raspberry Pi realiza coleta, salva os dados, controla a interface com o usuário e realiza conexão à *Web* via *Ethernet*.

Em 2018, Fakih, Naik, Rais e Maniar (2018) desenvolveram um projeto baseado na tecnologia de automação residencial e sistema de geração de energia fotovoltaica, em que foi utilizado o Arduino como microcontrolador e a plataforma *Blynk*®, aplicando o conceito de IoT. Obteve-se o controle das cargas através dos relés controlados pelo Arduino e a visualização dos status, assim como o comando através do aplicativo *Blynk* instalado no dispositivo móvel.

Khairnar e K (2018) desenvolveu um protótipo cujo objetivo era monitorar e controlar um motor trifásico de indução, sendo utilizado o ADE7758 para obtenção dos parâmetros elétricos do motor.

Xu e Wang (2019) aplicaram os *softwares Labview*® e *Matlab*®, em conjunto com um sistema de monitoramento de energia com o ADE7758, de modo a analisar a qualidade de energia e previsão de carga em tempo real, monitorando a saúde e o desempenho operacional do equipamento, obtendo efetividade nas interferências causadas por surtos de curta duração e monitoramento inteligente.

Em Jadhav e Malode (2019), foi desenvolvido um servidor de mídia *offline* de baixo custo, utilizando o *Raspberry Pi* como servidor, que possibilitou o acesso dos dados sem a necessidade do servidor *on-line*, podendo ser utilizado para outras aplicações, como vídeos, *e-books* e outros.

Já em Ramalingam, Baskaran e Kalaiarasan (2019), foi desenvolvido um sistema de monitoramento e controle da poluição do ar, onde foi utilizado o *Raspberry* para processamento e disponibilização dos dados através página da *Web* e servidor local do *Blynk*®. Obteve-se assim um sistema de monitoramento contínuo bastante útil e confiável, com um baixo custo e uma taxa de resposta rápida em relação aos métodos manuais utilizados anteriormente.

Patil, Patil e Khude (2020) desenvolveram um sistema de fornecimento de energia pré-pago, onde o consumidor monitora o consumo de energia elétrica e caso queira suspender o fornecimento, pode ser realizado a distância via IoT. Foi empregado o Circuito Integrado (CI) ADE7758 para realizar a medição elétrica. Obteve-se assim um medidor em que o consumidor

é cobrado de acordo com a energia elétrica necessária, controlando o consumo e com o foco em redução do desperdício.

### 3. MATERIAIS E MÉTODOS

Neste capítulo são apresentados os aspectos relacionados ao desenvolvimento do sistema, os componentes, funções destes e métodos para construção do protótipo e sistema supervisor.

A proposta deste projeto é realizar o monitoramento e aquisição de dados elétricos de uma usina de geração fotovoltaica, disponibilizando os dados para o usuário por meio de um ambiente gráfico simples e intuitivo, utilizando equipamento de comunicação móvel.

#### 3.1 Local da pesquisa

A pesquisa foi desenvolvida na Universidade Federal Rural do Semi-Árido (UFERSA), Campus Leste, localizado na cidade de Mossoró-RN, em um sistema de geração de energia fotovoltaica (Mossoró I) situado no Centro Integrado de Inovação Tecnológica do Semiárido (CITED), que foi instalada com fins didáticos, possibilitando os testes e desenvolvimento deste projeto.

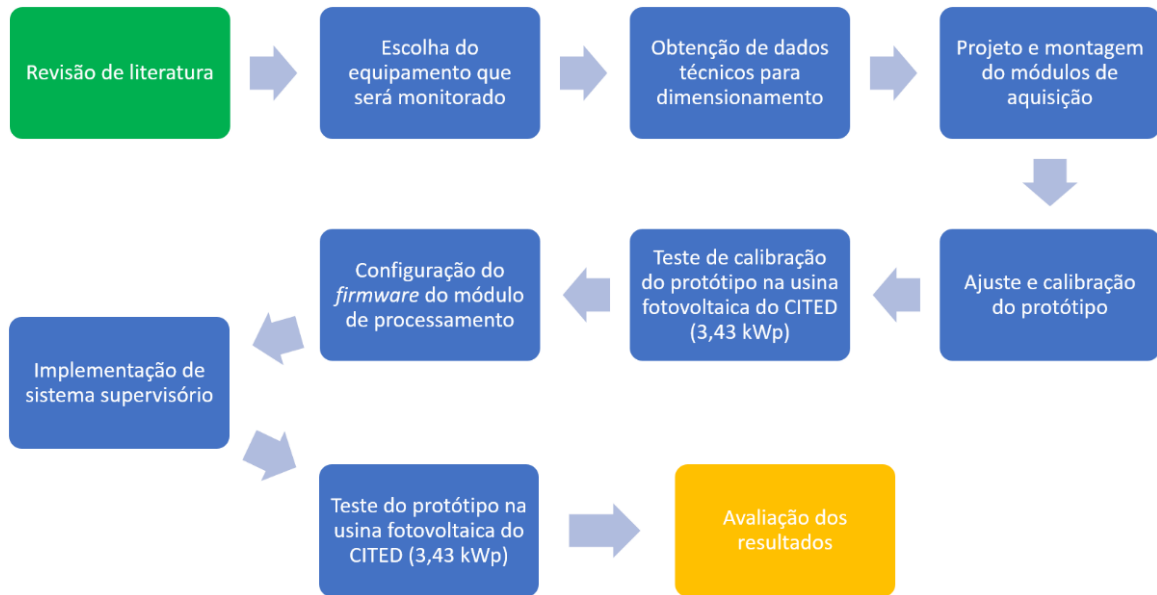
A usina Mossoró I, possui 14 módulos do fabricante *Yingli Solas*, modelo YL245P-29b, cuja potência de pico total é de 3,43 kWp (quilo-watt pico). Os módulos são instalados em uma estrutura no solo, por trás do prédio do CITED, sendo posicionado no azimute 0° com inclinação de 11° (UFERSA, 2019).

O seu inversor de frequência instalado é abrigado no laboratório de sistemas de energia, de fabricante SMA, modelo *Sunny Boy 3000HF-US*, que possui corrente nominal de saída à 240V (Volts) de 12,5A (Ampères) e a corrente máxima para 208V é de 14,5A, sendo esses valores utilizados para dimensionamento do protótipo (SMA SOLAR TECHNOLOGY©, 2020).

### 3.2 Procedimento da pesquisa

Para desenvolver o projeto proposto, o mesmo foi dividido em etapas, a fim de se obter uma visão mais clara de cada etapa e, por consequência, uma melhor organização e otimização. Tais etapas serão descritas ao decorrer deste tópico e representadas de forma resumida na Figura 3.

Figura 3: Fluxograma com etapas para realização do trabalho de dissertação.



Fonte: Autoria própria.

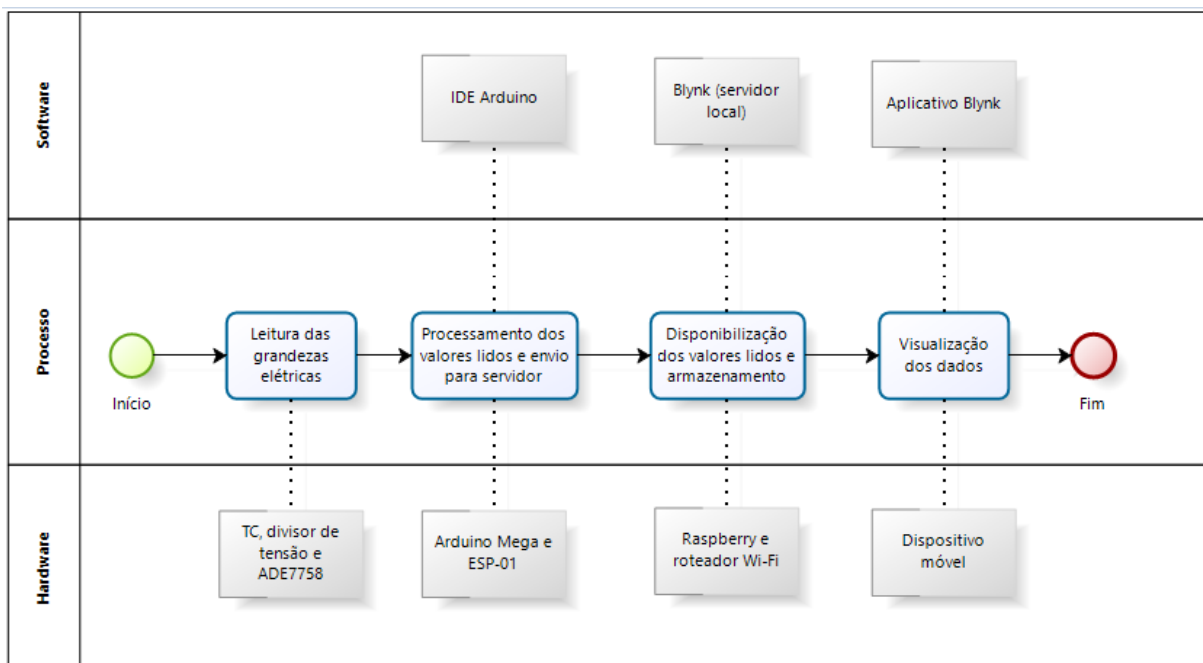
- Foi escolhida a usina fotovoltaica da UFERSA (Mossoró I), por se tratar de uma usina com fins didáticos e fácil acesso para realizar os testes e validação do projeto;
- Obtenção de dados técnicos relacionados à usina fotovoltaica da UFERSA (Mossoró I), sendo eles: potência nominal, corrente e tensão de saída;
- Projeto, montagem e calibração do módulo de aquisição de dados, que será responsável pela medição dos parâmetros elétricos gerados pelo inversor monitorado;
- Teste de calibração do protótipo na usina fotovoltaica de modo a validar a calibração e ajustes;
- Implementação do *firmware* do módulo de processamento de dados, responsável pela leitura e processamento da informação e comunicação com o servidor remoto;

- Configuração do servidor local, compreendendo as etapas: configuração do servidor *broker*; integração do servidor local ao módulo de processamento de dados; integração da aplicação à interface do usuário;
- Teste do protótipo na usina fotovoltaica Mossoró I (3,43 kWp) e avaliação dos resultados.

### 3.3 Desenvolvimento do projeto

Para desenvolvimento do projeto proposto se fez necessário, visando uma melhor organização, dividi-lo em três partes (conforme mostrado no fluxograma na Figura 4), onde na primeira serão detalhados os componentes físicos utilizados (*hardware*), em seguida os *softwares* utilizados para implementar as funções necessárias e por último o processo como todo.

Figura 4: Representação global do protótipo.



Fonte: Autoria própria.



### 3.3.1 Hardwares utilizados

Para possibilitar a leitura dos parâmetros elétricos pelo usuário é necessário, primeiramente, de sensores que possam mensurar as grandezas elétricas, de microcontroladores para processar os valores e disponibilizar os dados para um dispositivo em que possam ser lidos os mesmos. Essas funções são realizadas pelos *hardwares* implementados no projeto, sendo os sensores de corrente descritos na seção a seguir.

#### 3.3.1.1 Transformador de Corrente (TC)

O transformador de corrente transforma a corrente que circula no enrolamento primário, através de um acoplamento magnético, em uma corrente induzida no secundário de menor valor proporcional a corrente do primário, possibilitando assim ler valores altos de correntes com instrumentos de menor capacidade.

Na Figura 5 é mostrada a imagem do TC utilizado no projeto, que é do tipo núcleo dividido (não invasivo), o que permite a conexão e desconexão do circuito primário sem interrupção. O mesmo possui relação de transformação de 0-30 A no primário, com saída 0-1V, acurácia (X) de  $\pm 1\%$  e erro ( $\epsilon$ )  $\leq 0,2\%$ , podendo ser utilizado como medição ou proteção (ALIEXPRESS©, 2020).

Figura 5: Imagem física do TC utilizado no projeto (Modelo: SCT013-030).



Fonte: Autoria própria.

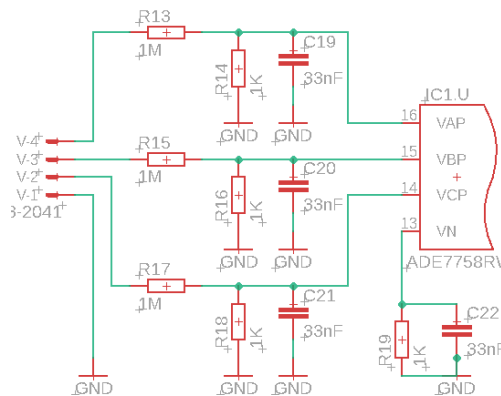
Mesmo com o TC modificando a grandeza de entrada do sinal de corrente para um nível compatível com o CI ADE7758, na entrada do sinal devem ser instalados resistores e capacitores. O intuito é atenuar ruídos de acordo com o desenho esquemático que será demonstrado na subseção divisor de tensão.

### 3.3.1.2 Divisor de tensão

Semelhante ao TC, deve-se adequar o sinal de entrada da tensão de modo a ficar na faixa de leitura do CI ADE7758 ( $\pm 500$  mV), para isso, será utilizado um divisor de tensão (semelhante ao que se encontra no *datasheet* do CI), que consiste em uma associação de resistores de modo que a tensão de saída seja proporcional a tensão de entrada, porém, em um nível compatível ao CI.

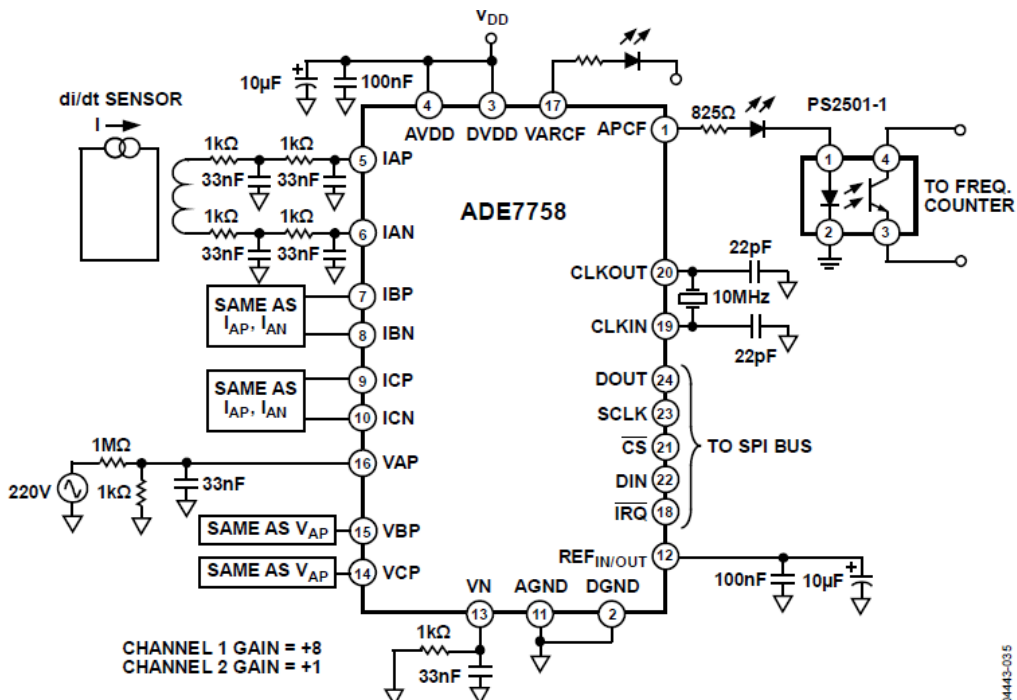
Nas Figuras 6 e 7, são mostrados, respectivamente, o divisor de tensão implementado (desenho feito no *software Eagle*) e o esquema de ligação proposto no *datasheet* do CI ADE7758.

Figura 6: Divisor de tensão implementado.



Fonte: Autoria própria.

Figura 7: Esquemático de ligação do CI ADE7758.



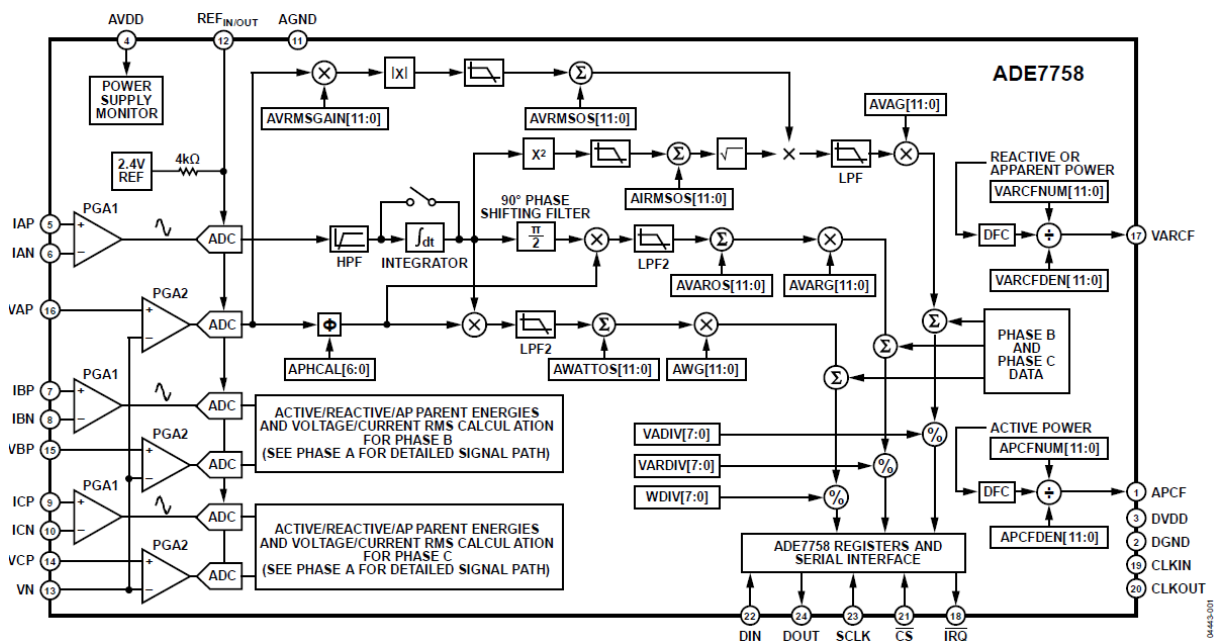
Fonte: ANALOG DEVICE©, 2019.

Com todas as ligações realizadas de acordo com o *datasheet*, desenho mostrado na Figura 7, e as entradas de sinais com valores de entrada compatível com o CI (o valor máximo no sinal de entrada analógica é de  $\pm 500$  mV), será detalhado como é realizada a medição das grandezas elétricas e disponibilização dos dados lidos na subseção seguinte a respeito do CI ADE7758.

### 3.3.1.3 Circuito integrado ADE7758

O Circuito Integrado ADE7758 é um dispositivo de medição de energia elétrica trifásica, fabricado pela empresa *Analog Devices*®, que embarca conversores analógico-digitais, mecanismo de cálculo de parâmetros elétricos e interface serial. O mesmo foi projetado para mensurar, com precisão, energia ativa, reativa e aparente em várias configurações trifásicas, assim como tensão e corrente eficaz em RMS (*Root Mean Square*), tudo integrado em um único *chip*. A Figura 8, representa o diagrama em blocos do CI ADE7758 (ANALOG DEVICE®, 2019).

Figura 8: Diagrama de blocos ADE7758.



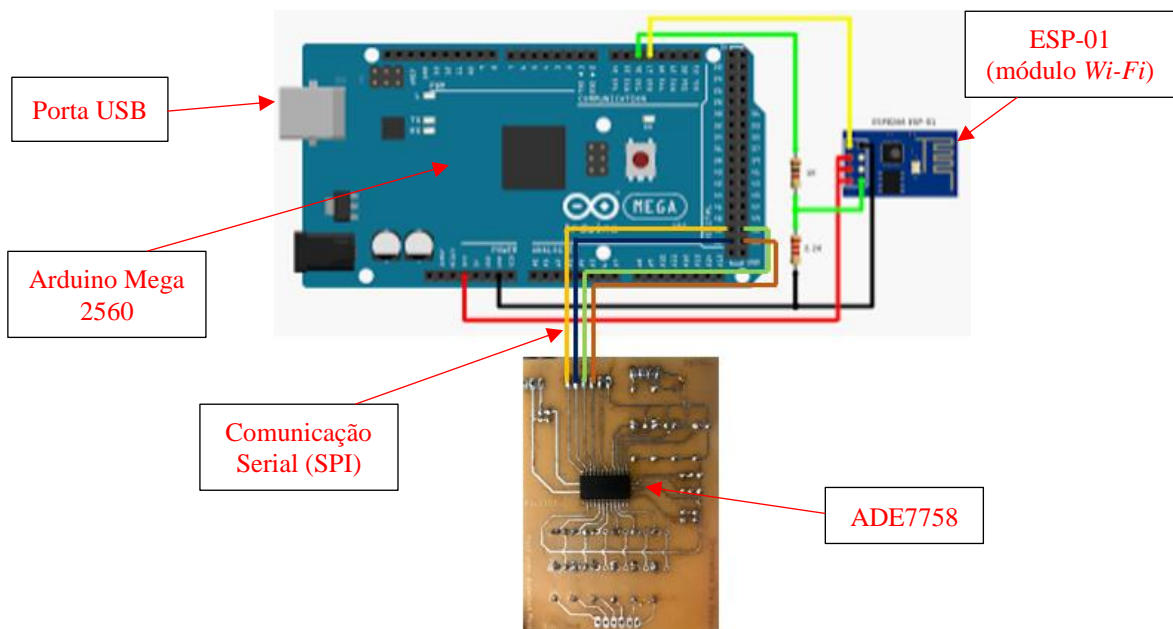
Fonte: ANALOG DEVICE®, 2019.

A comunicação do ADE7758 com o microcontrolador foi realizada através do protocolo SPI (*Serial Peripheral Interface*), por meio dos pinos DIN, DOUT, SLCK e CS, que por sua vez, se conectam aos pinos correspondentes do Arduino Mega 2560, conforme será mostrado na subseção a seguir.

### 3.3.1.4 Arduino Mega 2560

O Arduino Mega 2560 é uma placa de prototipagem cujo *chip* microcontrolador é baseado no ATmega2560, que possui interface *USB-Serial* que possibilita ser inserido o *firmware*. Na Figura 9, podem ser verificadas as ligações do Arduino Mega 2560, cuja conexão *USB-Serial* será utilizada para inserir o *firmware* e prover a alimentação do projeto. A comunicação com o módulo de aquisição de dados será realizada através do protocolo SPI com o ADE7758. A comunicação com a rede *Wi-Fi* será possível devido a comunicação que será feita com o ESP-01., através da interface que será detalhado na subseção ESP-01 (módulo *Wi-Fi*).

Figura 9: Esquemático de ligação entre: Arduino Mega 2560, ESP-01 e ADE7758



Fonte: Próprio autor.

### 3.3.1.5 ESP-01 (módulo *Wi-Fi*)

O módulo *Wi-Fi* ESP8266 modelo ESP-01 é utilizado para possibilitar que os microcontroladores tenham uma conexão *Wi-Fi* de maneira fácil e de baixo custo.

No projeto foi implementado o ESP-01 em conjunto com o Arduino Mega, que utiliza a comunicação através da via serial utilizando os pinos RX e TX em ambas as placas. Como o ESP-01 a tensão de alimentação e comunicação é com o nível de tensão de 3,3V, e as saídas digitais do Arduino Mega são de 5V, foi implementado um divisor de tensão para evitar danos ao ESP-01, conforme pode ser verificado na Figura 9.

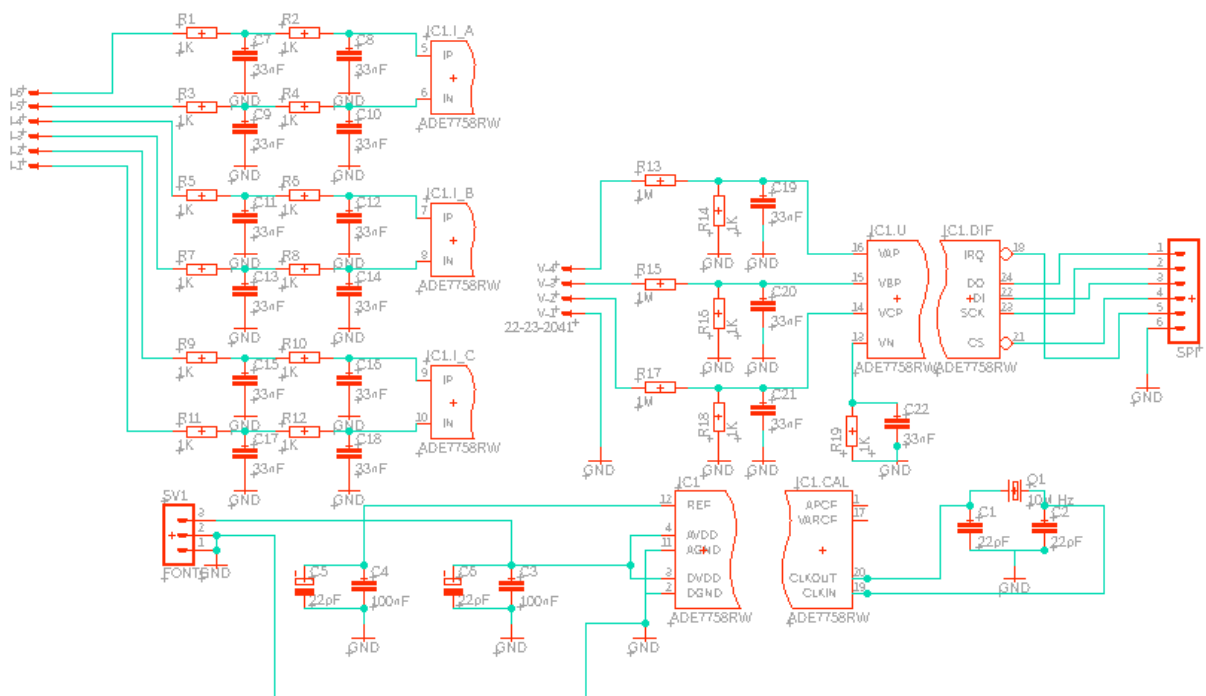
De modo a garantir a correta comunicação, adequada conexão e melhor acomodação dos componentes eletrônicos, foram desenvolvidas placas de circuito impresso para adequação do sinal e alimentação do ESP-01 e as devidas ligações já descritas do CI ADE7758, sendo elas descritas na subseção Placas de Circuito Impresso (PCI).

### 3.3.1.6 Placas de circuito impresso (PCI)

Para reduzir erros devido às falhas em conexão, e melhor entendimento do projeto, foram desenvolvidas duas placas de circuito impresso utilizando o *software Eagle* que é distribuído pela *CadSoft* (adquirida pela empresa *Autodesk* em 2016), onde para usuários estudantes ou/e *hobbyistas*, não é necessário a compra da licença, sendo utilizado a versão “*freeware*”.

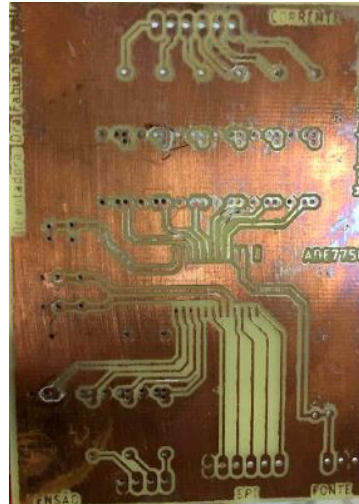
Devido à possibilidade da versão sem custos, há uma grande quantidade de bibliotecas disponíveis e componentes, o que reduz bastante o tempo de desenvolvimento do projeto. Foram confeccionadas duas placas, sendo uma **placa de medição** destinada a receber e adequar os dados provenientes do sistema a ser monitorado (no caso do projeto, será o inversor de frequência) e disponibilizar os dados através da saída SPI (Figura 10 e Figura 11); e uma **placa de comunicação Wi-Fi** com a finalidade de adequar os sinais de saída do Arduino Mega 2560 com o ESP-01 (Figura 12).

Figura 10: Esquema elétrico da placa de medição



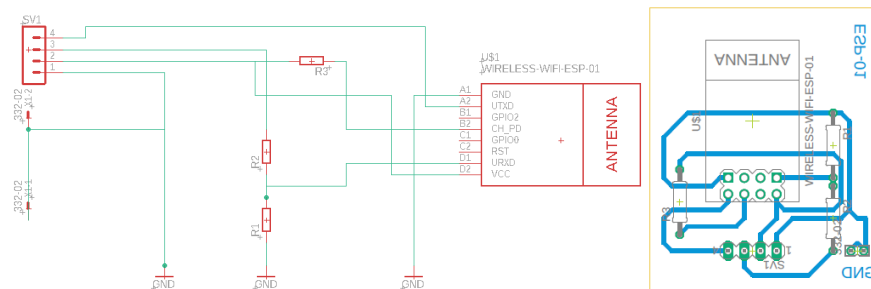
Fonte: Autoria própria.

Figura 11: *Layout* físico da placa de medição



Fonte: Autoria própria.

Figura 12: Diagrama elétrico e *layout* físico da placa de comunicação *Wi-Fi*.



Fonte: Autoria própria.

Com a implementação das PCI's, obteve-se uma maior confiabilidade nos módulos de aquisição de dados e processamento, sendo bastante relevante para que o *firmware* possa ler dados confiáveis e serem devidamente processados e enviados para o servidor local que se encontra instalado no *Raspberry*, que será mostrado na próxima subseção.

### 3.3.1.7 *Raspberry Pi*

O *Raspberry Pi3* foi utilizado como computador, por possuir tamanho reduzido e baixo custo em relação a um computador convencional, aliado ao baixo consumo de energia, além de possuir várias aplicações envolvendo internet das coisas. Ele possui saída de vídeo, conexão para áudio, quatro portas USB (*Universal Serial Bus*) e porta *Ethernet* para possibilitar conexão à rede (UPTON, 2013).

Por possuir baixo custo, baixo consumo de energia e poder de processamento, foi instalado no *Raspberry Pi3* o servidor local do *Blynk*®, de modo a possibilitar uma maior segurança, maior estabilidade e baixa latência. Foi conectado numa rede *Wi-Fi* dedicada para

obter todos os benefícios e a facilidade de instalação do servidor local em qualquer local que possua acesso a esta rede.

#### 3.3.1.8 Roteador *Wi-Fi*

Para possibilitar a comunicação entre os dados lidos pelo Arduino Mega 2560, o servidor do *Blynk*® instalado no *Raspberry Pi3* e o dispositivo móvel para visualização do usuário, foi configurado um roteador dedicado de fabricante TP-LINK®, modelo TL-WR740N.

No sentido de ampliar a segurança dos dados do projeto, foi implementado IP (*Internet Protocol*) fixo ao MAC (*Media Access Control*) do *Raspberry*, assim como alterado o *login* da rede e a senha de acesso.

### 3.3.2 *Softwares utilizados*

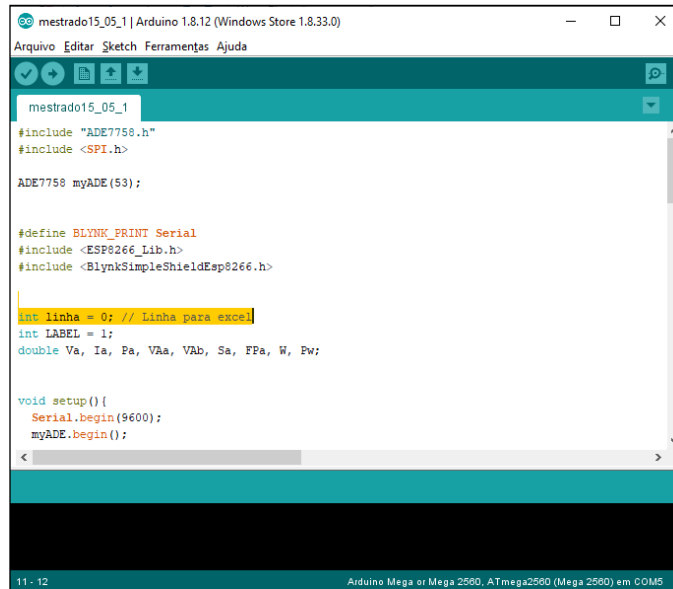
Após a aquisição dos dados realizada pelos sensores em conjunto com o ADE7758 é necessário realizar comunicação com o microcontrolador, processar os dados e disponibilizar os valores ao usuário, ao qual foi implementado um *firmware* no microcontrolador e uma plataforma para possibilitar a visualização e armazenamento dos dados, sendo estes detalhados nas subseções a seguir.

#### 3.3.2.1 Implementação do *firmware*

O conjunto de instruções para possibilitar a comunicação entre o ADE7758, Arduino Mega 2560, ESP-01, servidor local (*Raspberry Pi3*) e visualização nos dispositivos móveis, foi realizado utilizando a IDE (*Integrated Development Environment*) da plataforma de prototipagem eletrônica do Arduino, sendo utilizadas algumas bibliotecas pré-existentes e adequando à realidade do projeto.

É mostrado na Figura 13 a interface da IDE do Arduino, assim como o nome das bibliotecas utilizadas de modo a reduzir as linhas de comando, tornando o código mais claro e objetivo.

Figura 13: IDE Arduino.



```

mestrado15_05_1
#include "ADE7758.h"
#include <SPI.h>

ADE7758 myADE(53);

#define BLYNK_PRINT Serial
#include <ESP8266_Lib.h>
#include <BlynkSimpleShieldEsp8266.h>

int linha = 0; // Linha para exce:
int LABEL = 1;
double Va, Ia, Pa, VAb, VAb, Sa, FPa, W, Pw;

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

```

Fonte: Autoria própria.

Após programado, o Arduino Mega 2560 em conjunto com o ESP-01, será responsável por executar a requisição dos dados ao módulo de aquisição de dados, processar e disponibilizar os mesmos para a plataforma *Blynk*® em modo *looping*, que é descrito na subseção a seguir.

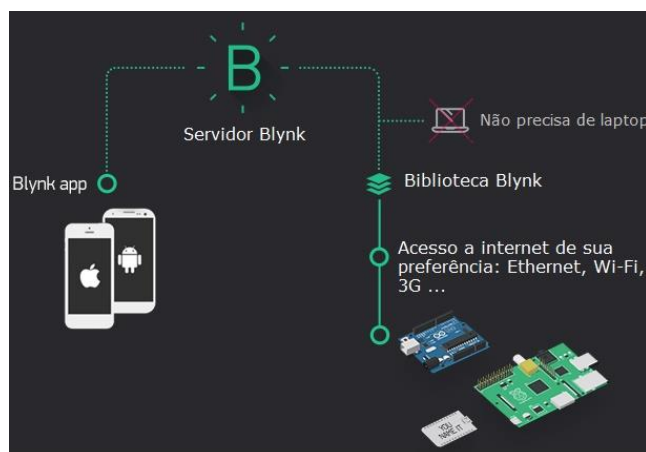
### 3.3.2.2 Plataforma *Blynk*®

A plataforma *Blynk*® permite realizar o controle e/ou monitorar um hardware programável através de aplicativo instalado no dispositivo móvel (*smartphone* ou *tablet*). Ele disponibiliza uma vasta gama de possibilidades de implementação em projetos de IoT, visto que o mesmo suporta placas de desenvolvimento como *Arduino*, *Raspberry Pi* e microcontroladores similares (SENEVIRATNE, 2018).

A plataforma *Blynk* é composta por três componentes principais que são mostrados na Figura 14.



Figura 14: Componentes da plataforma *Blynk*.



Fonte: Modificado de Blynk®, (2020).

- *Blynk app*: aplicativo instalado no dispositivo móvel em que é criada a interface com o usuário (supervisório).
- *Servidor Blynk*: responsável pela transmissão de dados entre o aplicativo *Blynk* e o *hardware*. O mesmo possui código aberto, desenvolvido em Java.
- *Biblioteca Blynk*: possibilita comunicação do servidor e processar todos os comandos entre o *Blynk app* e o *hardware*.

Assim como citado, o servidor do *Blynk* é responsável pela comunicação entre o dispositivo móvel e o *hardware*. Podendo ser na internet do *Blynk (Blynk Cloud)* ou num servidor local (rede privada), sendo que ambas as aplicações apresentam vantagens e desvantagens. O servidor *Blynk Cloud*, tem como principal vantagem o acesso ao supervisório em qualquer lugar, desde que esteja conectado à *Internet*. Contudo, não se tem garantia que o servidor permaneça online sempre. Já o servidor local, possui garantia de uma maior confiabilidade e baixa latência. No entanto, o acesso só é possível na área de abrangência da rede em que o mesmo está conectado (roteador).

### 3.3.3 Processo do projeto

Esta subseção, tem como objetivo integrar os recursos implementados no projeto de modo a ficar claro como os mesmos funcionam. Na subseção 3.3.3.1, é descrito o processo de leitura das grandezas elétricas e sua interface.

### 3.3.3.1 Leitura das grandezas elétricas

Os sensores de tensão e corrente adequam o sinal das grandezas elétricas para um valor compatível da entrada de sinal analógica do CI ADE7758, onde é realizado o processamento dos sinais analógicos de modo a obter os valores digitais das grandezas elétricas. Os valores obtidos são disponibilizados através do protocolo de comunicação SPI que se encontra conectado diretamente com o Arduino Mega 2560, que irá realizar o processamento e envio para o servidor, conforme será descrito na subseção 3.3.3.2.

### 3.3.3.2 Processamento dos valores lidos e envio para o servidor

A porta de comunicação SPI do Arduino Mega 2560 é conectado diretamente com o CI ADE7758, onde são obtidos os valores das grandezas elétricas lidas pelo CI ADE. Para isso, foi implementado no *firmware* a biblioteca do ADE7758.h que se encontra em acesso público, desenvolvido por Auguste (2019), sendo necessárias algumas adaptações e inclusão de funções para leitura de outras variáveis indispensáveis para o projeto, podendo ser verificado no Apêndice A deste trabalho.

O valor que é lido pelo ADE7758 é um valor proporcional ao valor de entrada, sendo necessário realizar a conversão do valor lido pelo ADE7758 e o valor real das grandezas elétricas. Para isso, são obtidas as equações de conversão através do método de calibração que será demonstrado subseção 3.3.4. Na Equação 1 e Equação 2, exemplificam-se as equações de tensão e corrente utilizadas no projeto.

$$\text{Tensão } A = 3,86 * 10^{-18} * x^3 - 9,46 * 10^{-12} * x^2 + 1,99 * 10^{-4} * x^1 - 1,57 \quad (\text{Equação 1})$$

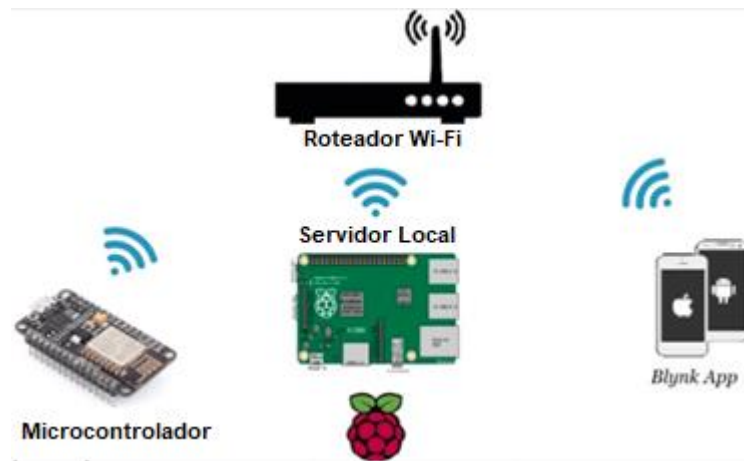
$$\text{Corrente } A = 4,59 * 10^{-22} * y^3 - 2,34 * 10^{-16} * x^2 + 5,67 * 10^{-7} * x^1 - 3,42 * 10^{-2} \quad (\text{Equação 2})$$

Depois de processados e devidamente quantificados, os dados são enviados para o supervisor através da comunicação *Wi-Fi*, que é possível devido a implementação do ESP-01 que se encontra ligado na porta de comunicação serial do Arduino Mega 2560 e o uso das bibliotecas no *firmware* (ESP-8266 e *Blynk*).

### 3.3.3.3 Disponibilização dos valores lidos e armazenamento

Com o objetivo principal de realizar a interface dos dados lidos pelo projeto com o usuário e armazenar os mesmos, foi implementada uma arquitetura semelhante à Figura 15, em que foi utilizado um *Raspberry Pi3* como servidor local e um roteador *Wi-Fi* dedicado, sendo diferente apenas o microcontrolador, que foi substituído pelo conjunto Arduino Mega e ESP-01.

Figura 15: Arquitetura utilizando *Raspberry Pi* como servidor local.



Fonte: Modificado de Forever, (2019).

Foi configurado o *Raspberry Pi3* como servidor local, que de acordo com *Blynk*® (2019), agrega ao projeto uma melhor segurança, estabilidade, baixa latência e máxima privacidade. Na Figura 16, é mostrado o servidor *Blynk*® ativo no *Raspberry*.

Figura 16: Iniciação do servidor local do *Blynk*®.

```

pi@raspberrypi: ~/Blynk
Arquivo Editar Abas Ajuda
pi@raspberrypi:~$ cd Blynk
pi@raspberrypi:~/Blynk$ java -jar server-0.41.12-java8.jar -dataFolder /home/pi/Blynk
Blynk Server successfully started.
All server output is stored in folder '/home/pi/Blynk/logs' file.

```

Fonte: Autoria própria.

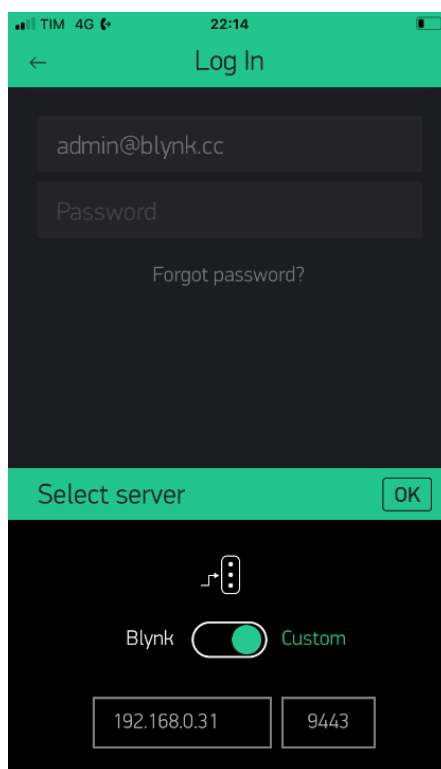
Após a instalação do servidor local e rede *Wi-Fi* dedicada, deve ser implementado no firmware do microcontrolador os dados do IP do servidor e da rede *Wi-Fi*, para que o mesmo se conecte ao servidor e que os dados fiquem disponível para serem visualizados através do *Blynk app*.

### 3.3.3.4 Visualização dos dados

O aplicativo do *Blynk*® instalado, tem como função principal a interface do projeto com o usuário e para que isso aconteça é necessário que estejam conectados na mesma rede *Wi-Fi* (servidor, *hardware* e *software*), que os dados da interface com o usuário sejam compatíveis com o servidor (IP e porta de comunicação do servidor) e dados do *firmware* (senha de autenticação do aplicativo ser a mesma do firmware, modelo da placa e modo de comunicação).

Na Figura 17 é mostrado a interface do aplicativo *Blynk*® no dispositivo móvel e os campos onde são inseridos o login e senha do usuário, o IP e porta de comunicação do servidor local que foi utilizado.

Figura 17: Configurações do aplicativo *Blynk*®, instalado no dispositivo móvel.

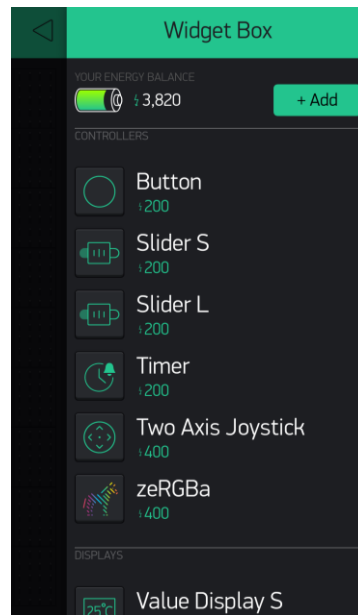


Fonte: Autoria própria.

Com todos os dados inseridos, é criado um novo projeto no aplicativo, e facilmente é criada a interface utilizando o “*Widget box*”, que são os itens de representação visual de como o dado será apresentado na interface do usuário na tela.

Quando se utiliza o servidor do *Blynk*® na internet, é limitada a quantidade *widget* de interface com o usuário de acordo com a “energia” que o mesmo representa, sendo cada demonstrado um exemplo na Figura 18.

Figura 18: *Widget Box* e balanço de energia.

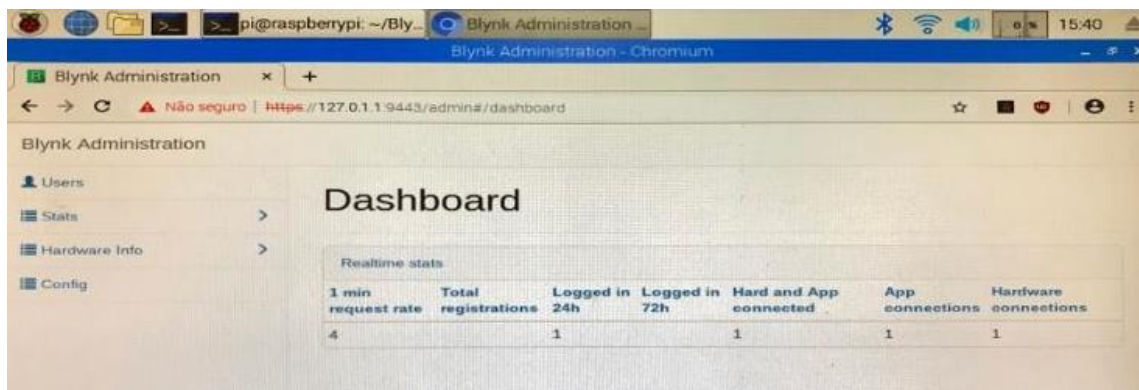


Fonte: BLYNK® (2020).

Cada item de interface apresenta um valor, porém ao utilizar servidor local não se tem problemas de limitação, visto que se pode configurar diretamente no servidor a energia necessária para a interface, ou deixar o valor padrão que é de \$100.000,00, que atende a interface utilizada neste projeto.

A confirmação do correto funcionamento de comunicação entre os dispositivos (*firmware*, servidor local e aplicativo), pode ser verificada diretamente no servidor do *Raspberry*, conforme é demonstrado na Figura 19.

Figura 19: Informações do status no servidor local *Blynk*®.



Fonte: Próprio autor.

Para que o valor apresentado na interface seja confiável, tem-se a necessidade de realizar a correta calibração do projeto, sendo a mesma explicitada na subseção 3.3.4.

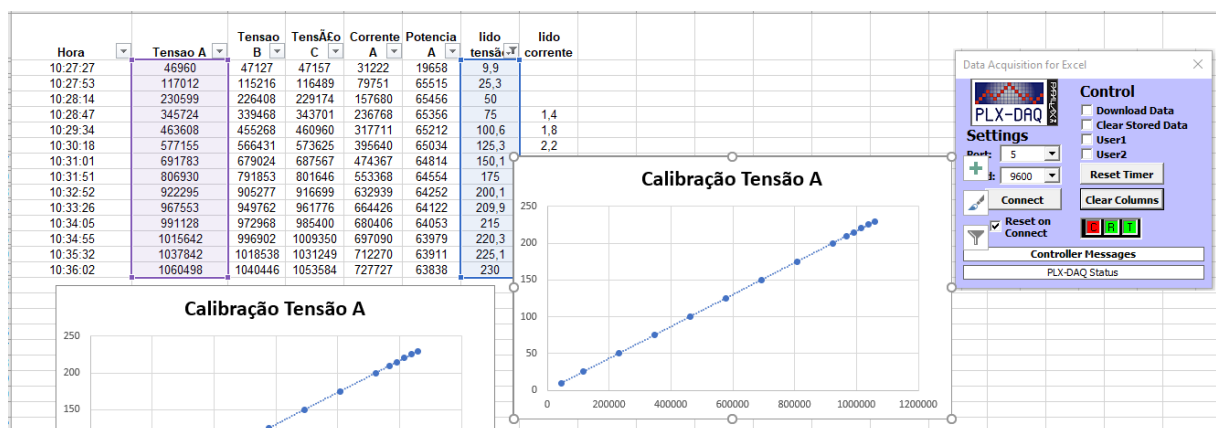
### 3.3.4 Calibração do projeto

A leitura das grandezas elétricas, realizada pelo módulo de aquisição de dados, mesmo sendo proporcional aos valores medidos, requer a realização da conversão para os valores reais. Para isso é necessária a obtenção das equações de conversão, sendo realizada a comparação dos valores lidos pelo *firmware*, com o valor medido por um instrumento que será utilizado como padrão. No caso desse projeto, o analisador de energia utilizado é do fabricante Minipa®, modelo: ET-5061C.

Também será utilizado um regulador de tensão monofásico, que possibilita variar os valores de tensão aplicada, obtendo a equação de tensão. Da mesma forma um banco de resistência será utilizado em conjunto com o regulador de tensão para que se possa obter a equação de corrente.

Para possibilitar a leitura dos valores lidos pelo *firmware*, foi utilizado o *software* PLX-DAQ (Parallax Microcontroller Data Acquisition for Excel), desenvolvido pela Parallax Copyright© 2007, possibilitando a comunicação serial dos dados lidos pelo Arduino Mega com o Excel, sendo utilizado para comparar o valor lido pelo *firmware* com o valor medido pelo analisador de energia conforme demonstrado na Figura 20. O código implementado para possibilitar a leitura pelo PLX-DAQ e os dados obtidos para a calibração estão contidos no Apêndice B.

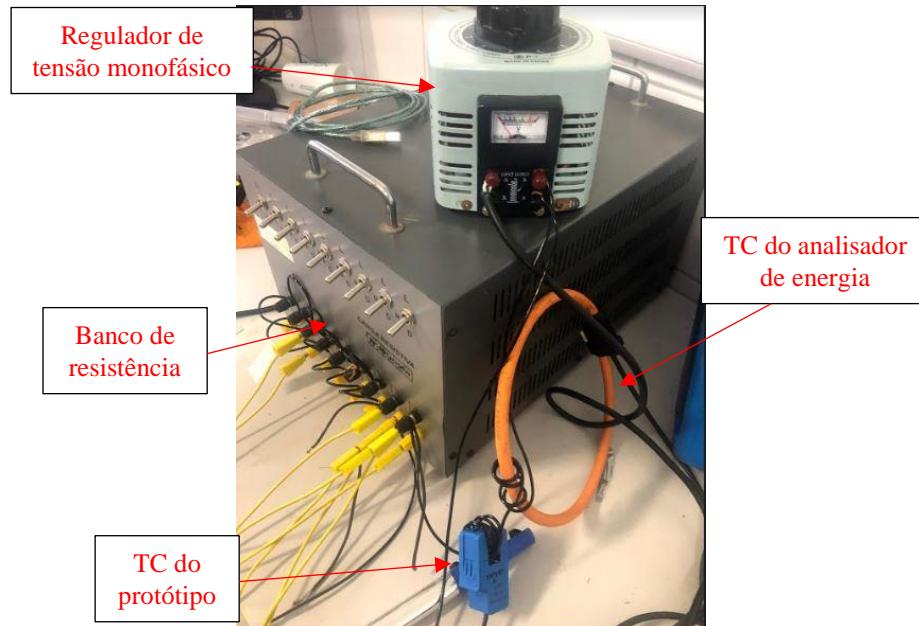
Figura 20: Valor lido pelo *firmware* com valor lido no analisador.



Fonte: Próprio autor.

Para realizar a calibração do protótipo, foram utilizadas todas as ferramentas descritas acima. Devido a limitação de potência da fonte de tensão, foram dadas 5 voltas nos TC's, para simular o valor da corrente nominal do inversor, como demonstrado na Figura 21.

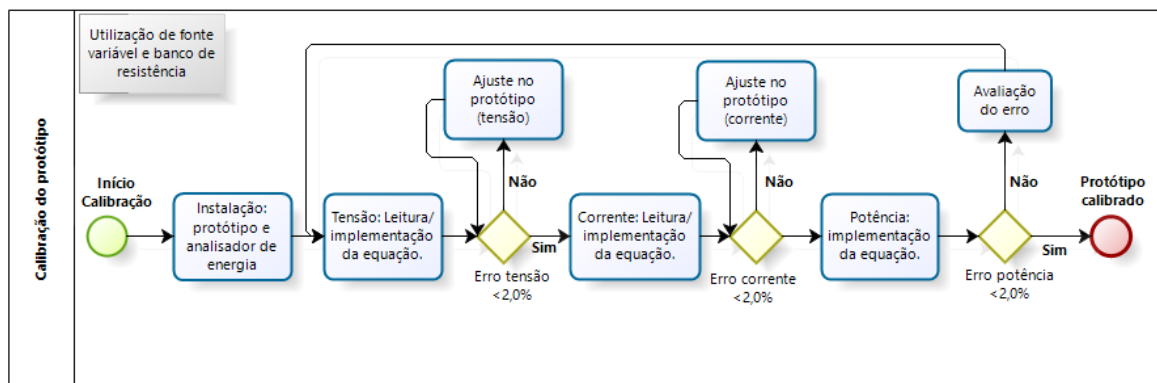
Figura 21: Regulador de tensão, banco de resistência e TC's.



Fonte: Próprio autor.

Com todos os instrumentos e equipamentos devidamente montados e funcionando adequadamente, seguiu-se a sequência de calibração conforme fluxograma demonstrado na Figura 22.

Figura 22: Fluxograma de calibração do projeto.



Fonte: Próprio autor.

Após a instalação dos equipamentos, foi realizada a leitura de tensão e corrente pelo *firmware* e no analisador de energia, obtendo as equações de ajuste de tensão (Figura 23) e corrente eficaz (Figura 24), sendo essas equações inseridas no *firmware* para que se possa seguir as avaliações dos erros.





Tabela 2: Erro de leitura do *firmware*.

Tensão (V)	Corrente (A)	Potência (kW)	Tensão Lida (V)	Corrente Lida (A)	Potência Lida (kW)	Erro Tensão (%)	Erro Corrente (%)	Erro potencia (%)
217,68	3,88	0,85	218	3,9	0,84	-0,15	-0,51	<b>1,19</b>
213,23	3,81	0,81	214,1	3,8	0,81	-0,41	0,26	0,00
205,33	3,67	0,75	206,7	3,7	0,75	<b>-0,66</b>	<b>-0,81</b>	0,00
224,90	4,0	0,90	225	4,0	0,90	-0,04	0,00	0,00
230,27	4,09	0,94	229,9	4,1	0,94	0,16	-0,24	0,00
206,31	7,48	1,54	207,2	7,5	1,55	-0,43	-0,27	-0,65
205,35	11,14	2,29	206,3	11,2	2,31	-0,46	-0,54	-0,87

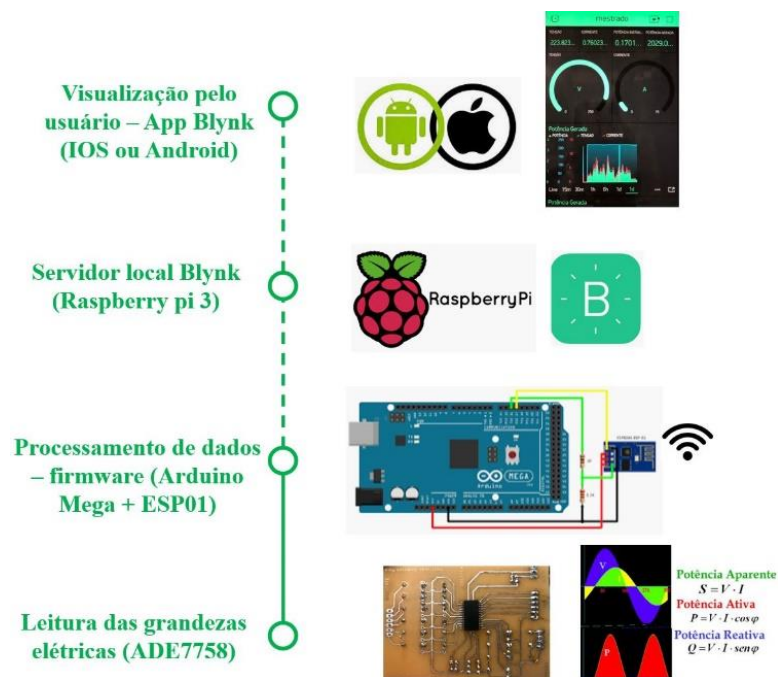
Fonte: Próprio autor.

Como pode ser visto na Tabela 2, o erro máximo de potência obtido foi menor que 2%, e o projeto foi considerado calibrado, habilitando o mesmo a seguir para a fase de testes e validação, conforme será detalhado no Capítulo 4 deste trabalho.

#### 4. RESULTADOS E DISCUSSÕES

O presente capítulo tem como objetivo principal mostrar os resultados obtidos após os testes do protótipo em campo, sendo detalhados os testes realizados, os critérios de validação utilizados e os resultados obtidos. Para facilitar o entendimento do projeto desenvolvido, na Figura 25 é mostrado a representação global, em que as leituras e processamento realizado pelo ADE7758 em conjunto com o Arduino Mega, fazendo parte da camada de percepção. O ESP01 junto com o *Raspberry Pi3* compõe a camada de rede, e por fim o aplicativo *Blynk*® instalado no dispositivo móvel executa o papel da camada de aplicação

Figura 25: Processos realizados pelo projeto.



Fonte: Autoria própria.

Foram realizados testes de modo a validar o protótipo desenvolvido, descrito na seção 4.1, onde é possível verificar como os testes foram realizados e seus resultados. Posteriormente foi realizada a avaliação final dos resultados obtidos.

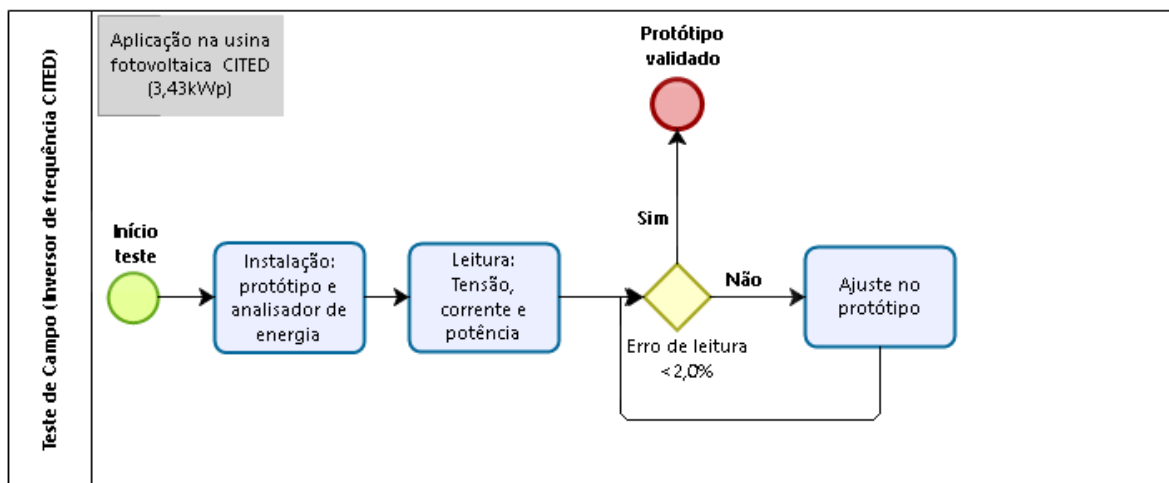
## 4.1 Testes e validação

Com o intuito de validar o protótipo desenvolvido, serão realizados dois testes, onde o primeiro irá validar a calibração do protótipo em campo (camada de percepção) e o segundo irá validar a comunicação e armazenamento de dados no servidor local (camadas de rede e aplicação). Na subseção 4.1.1, será detalhado como foi realizado o teste de acurácia e seu resultado.

### 4.1.1 Teste de acurácia na usina fotovoltaica do CITED (calibração)

Após a conclusão da calibração é iniciada a primeira etapa de validação do protótipo desenvolvido, onde é realizado o primeiro teste de campo, sendo o projeto instalado na Usina fotovoltaica Mossoró I, conforme já mencionado no Capítulo 3 deste trabalho, e seguindo o fluxograma que está representado na Figura 26.

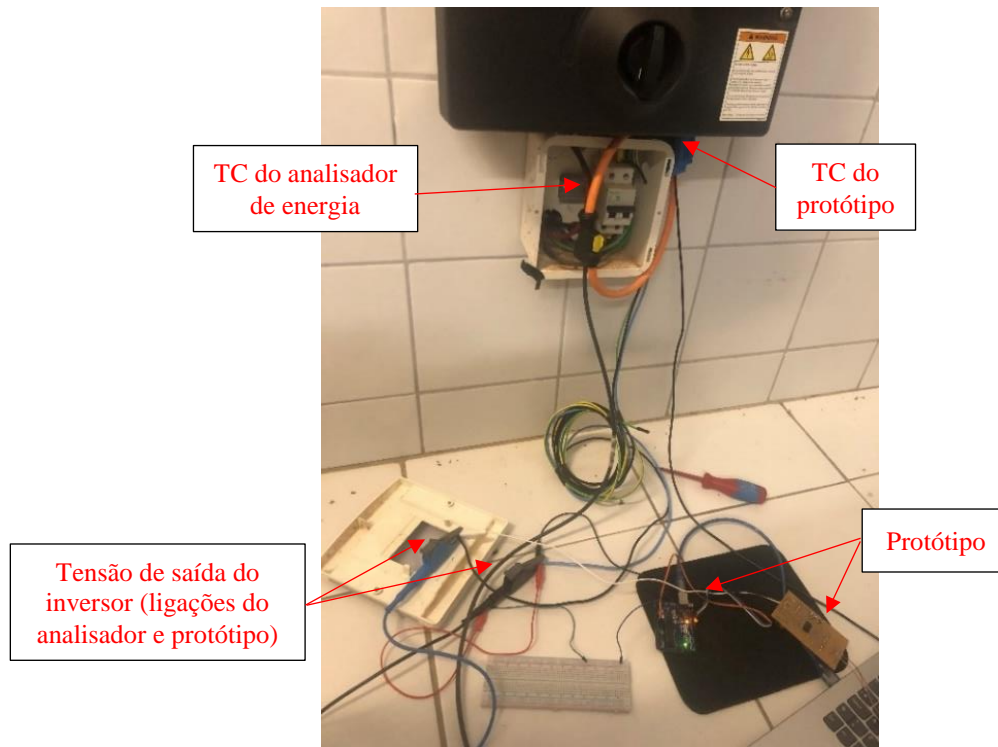
Figura 26: Fluxograma de teste de campo (calibração).



Fonte: Próprio autor.

Como mostrado na Figura 26, a instalação do protótipo e analisador de energia é realizada na saída do inversor da usina fotovoltaica, onde foram obtidos dados da tensão, corrente e potência gerada, lidos pelo protótipo e pelo analisador de energia, com o intuito de validar a acurácia do projeto. Foi instalado os TC's do analisador de energia e protótipo, assim como a medição de tensão (destacado na Figura 27).

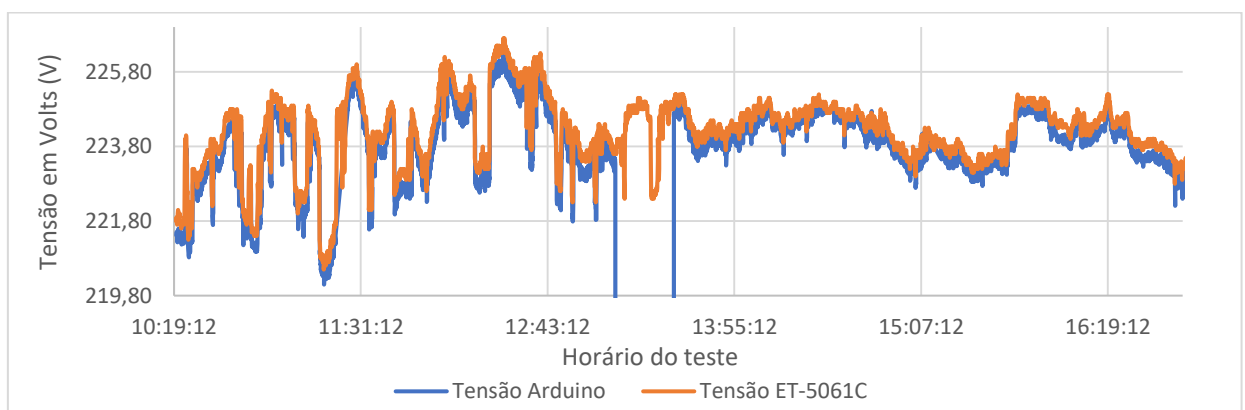
Figura 27: Ligações para medição do protótipo e analisador de energia.



Fonte: Próprio autor.

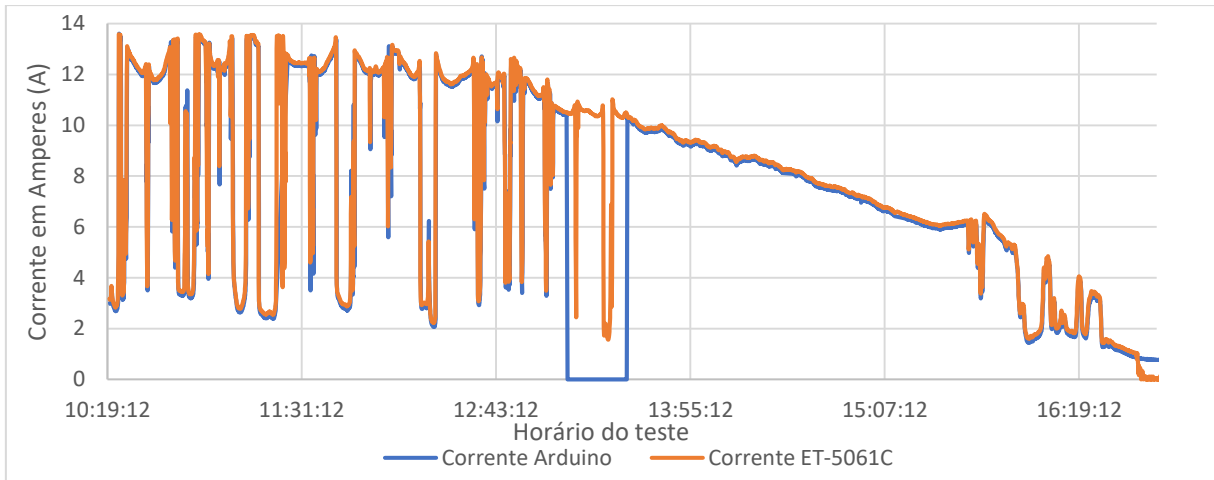
Após realização do teste, foram avaliados os dados armazenados no analisador de energia (ET-5061C) e proveniente do protótipo (Arduino). Para uma melhor avaliação, foram plotados gráficos dos valores lidos da tensão (Figura 28), corrente (Figura 29) e potência (Figura 30).

Figura 28: Leitura da tensão do protótipo e analisador de energia.



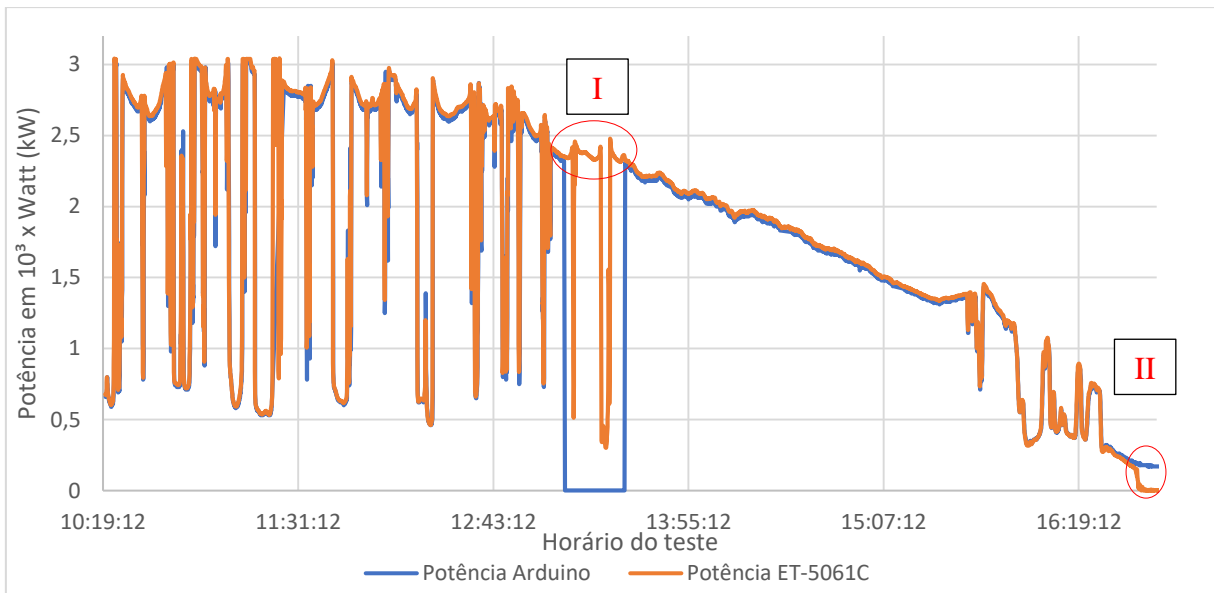
Fonte: Próprio autor.

Figura 29: Leitura da corrente do protótipo e analisador de energia.



Fonte: Próprio autor.

Figura 30: Leitura da potência do protótipo e analisador de energia.



Fonte: Próprio autor.

De acordo com os gráficos mostrados nas Figuras 28, 29 e 30, pode-se verificar que apenas em dois pontos os valores medidos pelo protótipo diferem do valor do analisador de energia, sendo as seguintes causas:

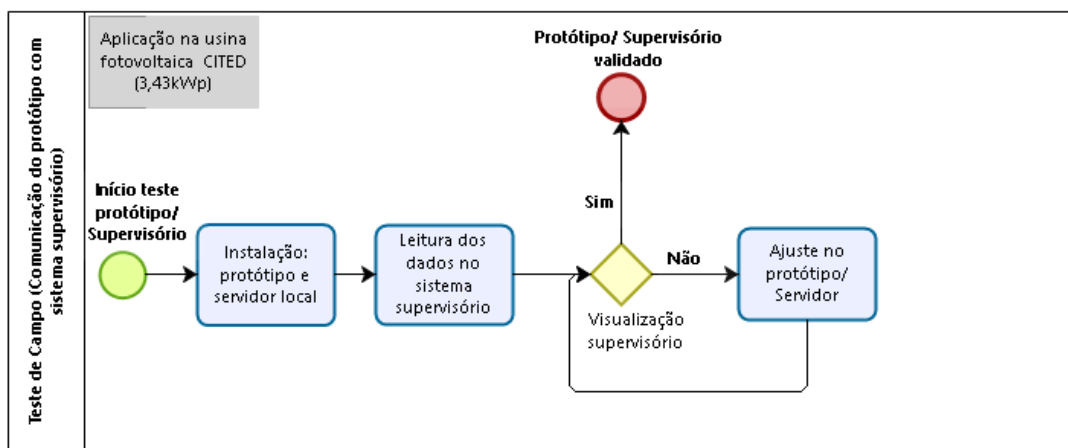
- I – Durante a leitura dos dados via *software* PLX-DAQ, às 13:10h o computador entrou em modo de hibernação, parando assim a comunicação. Foi verificado às 13:30h, e houve retorno da comunicação após sair do modo hibernação;
- II – Conforme a folha de dados do analisador de energia, para valores de corrente a baixo de 1A, são considerados zero pelo instrumento, sendo confirmado ao ler o valor da corrente no inversor a baixo de 1 A (MINIPA©, 2020).

O teste de campo de calibração foi considerado válido, habilitando o protótipo a ir para o teste seguinte, onde será avaliada a comunicação com o sistema supervisor, sendo este descrito na subseção 4.1.2.

#### 4.1.2 Teste de comunicação do protótipo (monitorando Usina Fotovoltaica)

Após a conclusão do teste de calibração foi iniciado o teste de comunicação, etapa em que será avaliada a comunicação do protótipo com o servidor e aplicativo no *smartphone*, assim como o armazenamento de dados no servidor local. Será seguido o fluxograma que está representado na Figura 31.

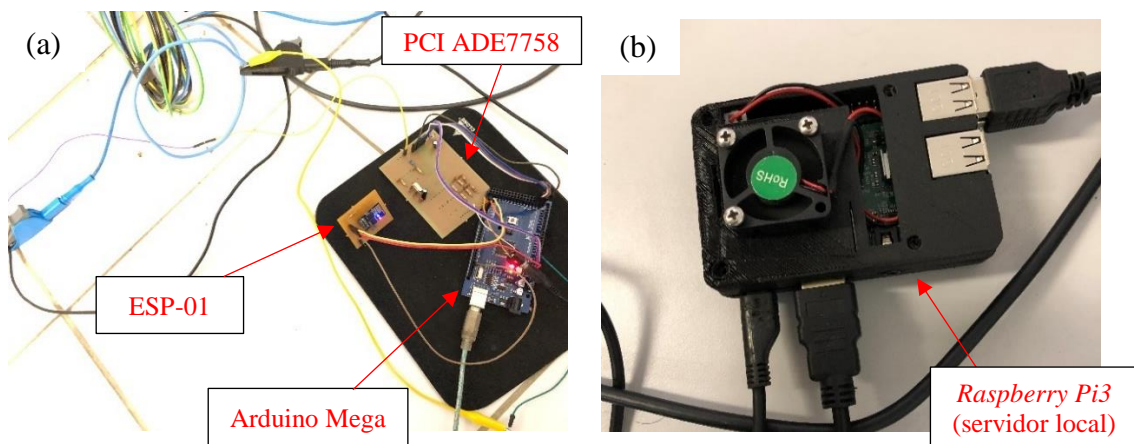
Figura 31: Fluxograma de teste de campo (comunicação com supervisor).



Fonte: Próprio autor.

Realizada a instalação do protótipo completo e devidamente configurado (Figura 32), é iniciado o teste de comunicação, onde são obtidos os dados de tensão, corrente e potência gerada instantânea, bem como o valor de potência gerada acumulada.

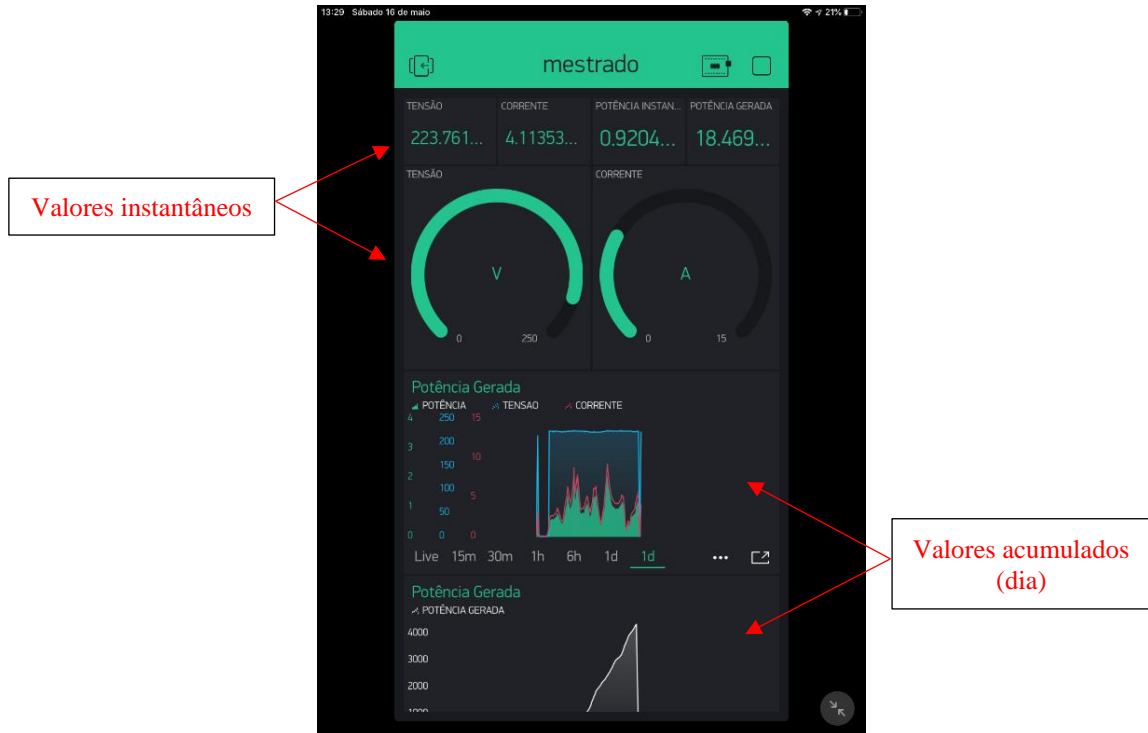
Figura 32: Placa com ADE7758, Arduino Mega e ESP-01 (a) e *Raspberry Pi3* (b).



Fonte: Próprio autor.

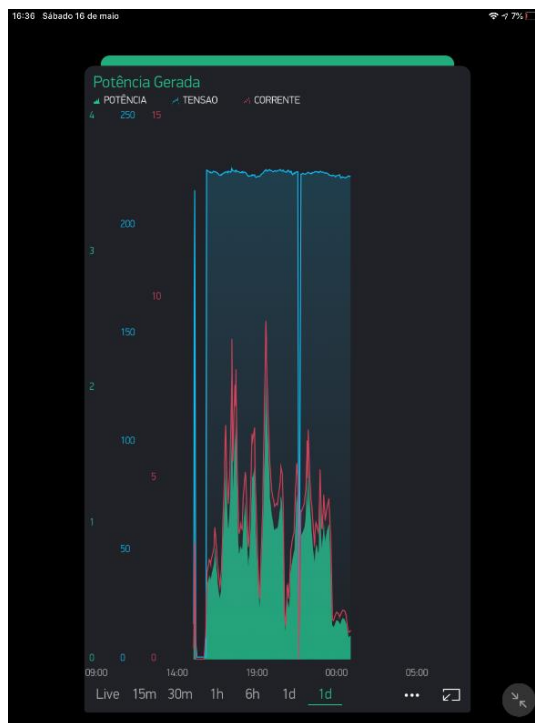
Durante o teste, foram avaliados os valores apresentados no aplicativo do *Blynk*® instalado, como também do arquivo armazenado no servidor local dos dados lidos, sendo apresentados nas Figuras 33 e 34.

Figura 33: Interface com o usuário marcando valores instantâneo e acumulado.



Fonte: Próprio autor.

Figura 34: Gráfico dos valores lidos de tensão, corrente e potência.



Fonte: Próprio autor.

Nos registros mostrados na Figuras 33 e 34, são utilizados o horário do servidor. Como o horário do servidor no início do teste encontrava-se diferente do horário local (cerca de 5 horas de diferença), os registros tiveram os valores deslocados, como pode ser verificado na Figura 34. Ao clicar no menu de configurações, representado pelo ícone de “três pontos” (*vide* Figura 34), é possível gerar o log das variáveis que estão sendo monitoradas no servidor local, sendo os mesmos registrados num intervalo de 60.000 milissegundos (para reduzir o tamanho dos arquivos gerados), validando assim o teste.

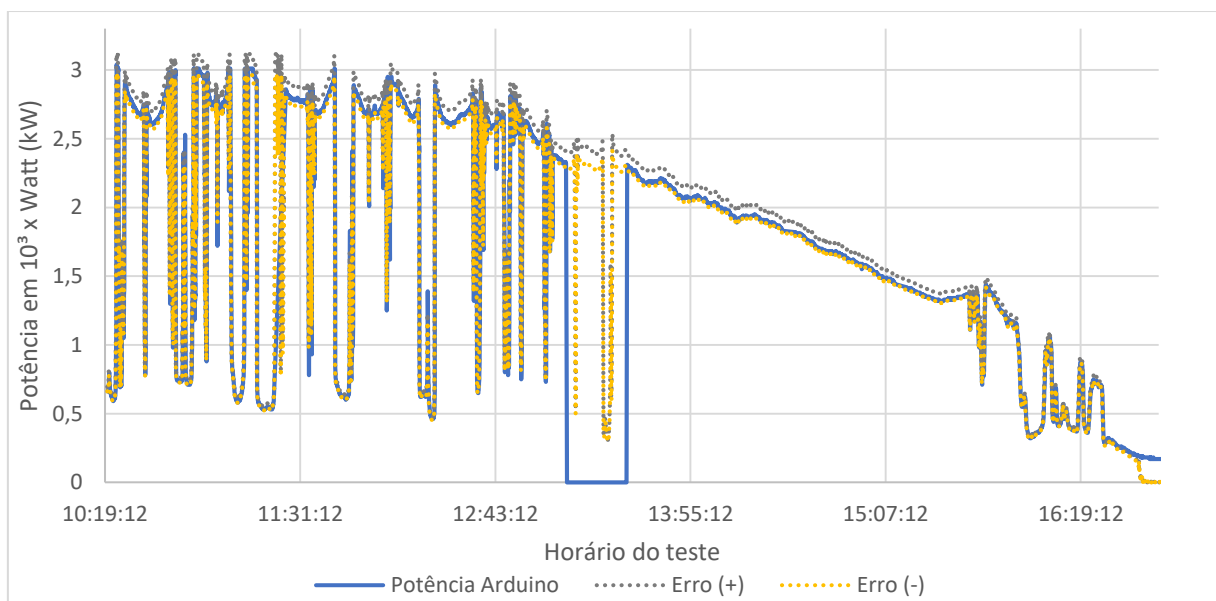
#### 4.2 Avaliação dos testes

De acordo com INMETRO (2012), para definir a classe de exatidão de um medidor eletrônico são realizados ensaios com 10% e 100% da corrente nominal ( $I_n$ ). Os medidores que apresentam erros percentuais em energia ativa de até 2,5% são classificados como classe A.

Durante o teste de calibração do protótipo o erro máximo obtido foi de 1,19%, mas devido às limitações do regulador de tensão e banco de carga, não foi possível realizar todos os testes que são descritos nos ensaios de verificação periódica para medidores eletrônicos que são realizados para medidores eletrônicos de acordo com INMETRO (2012).

Durante o teste de campo, foi verificado que a medição do protótipo na maior parte dos dados não ultrapassa a faixa de erro de  $\pm 2,5\%$ , conforme pode ser verificado na Figura 35.

Figura 35: Valor medido pelo protótipo e faixa do erro ( $\pm 2,5$ ).



Fonte: Próprio autor.



Por se tratar de um protótipo e das limitações de recursos para realizar a calibração e limitações de circuito e dos componentes, destaca-se por se manter boa parte do tempo dentro da faixa de erro de 2,5%, como pode ser constatado na Figura 35. Apesar das notáveis variações da geração, não houve grandes discrepâncias, estando dentro da faixa dos valores considerados como aceitáveis de acordo com INMETRO (2012) em boa parte do teste, portanto, o mesmo foi considerado válido.

Os resultados do teste de comunicação já apresentados nas Figuras 33 e 34 evidenciaram a comunicação do protótipo com o sistema supervisor, exibindo os dados elétricos em tempo real e armazenando o histórico dos mesmos, podendo ter acesso através do arquivo gerado no servidor. Sendo assim o projeto desenvolvido foi considerado validado.

## 5. CONCLUSÃO

Com a finalização do protótipo e realização dos testes na usina solar fotovoltaica Mossoró I, foi possível perceber que os principais objetivos definidos para este trabalho foram atingidos. O teste de acurácia realizado mostrou que o protótipo possui uma precisão aproximada de 2,5%, valor este equivalente a medidores eletrônicos com classe de exatidão A, e durante avaliação da calibração obteve-se um erro máximo de algo em torno de 1,2% em relação ao analisador de energia ET-5061C, ou seja, atingindo um valor relativamente próximo.

O servidor *Blynk*® executou com eficiência seu papel, demonstrando robustez, com um tempo de resposta desejável na atualização dos dados no aplicativo instalado nos equipamentos móveis. A possibilidade de fácil integração do mesmo servidor local para mais de um projeto é ideal para implementação de monitoramento de sistema de geração de energia fotovoltaica que possua mais de um inversor de frequência, como é o caso da usina solar Mossoró II, que possui 10 inversores (UFERSA, 2020).

No decorrer da elaboração do projeto, foram surgindo diversos obstáculos, entre eles, a reduzida quantidade de dados a respeito da utilização do ADE7758 como medidor de energia, onde as poucas bibliotecas existentes não obtinham comunicação ou ausência de parâmetros elétricos, sendo utilizada a biblioteca que continha uma maior quantidade de dados e inseridos novos parâmetros para que pudesse implementar no projeto.

Durante o processo de comunicação do CI ADE7758 com o Arduino, foram construídos três protótipos em placa universal, onde foi verificado que ocorria falha de conexão de componentes (principalmente o CI), o que ocasionava o erro de leitura ou não comunicação, sendo decidido como solução a construção de placa de circuito impresso. A primeira versão da PCI construída apresentava eventualmente distorções e leituras não lineares, sendo avaliado que o motivo era devido falha de referência, sendo solucionado após adicionar um plano de terra na PCI.

Outro grande obstáculo enfrentado foi a comunicação entre o CI ADE7758 e o microcontrolador ESP-32, que possui *Wi-Fi*, *Bluetooth* e duas portas de comunicação SPI, em que não se obteve sucesso na comunicação, em virtude da compatibilidade das tensões de comunicação do CI com o ESP-32 (CI-5V e ESP-32: 3,3V). Para contornar o problema, foi utilizado o Arduino Mega como microcontrolador e para obter a comunicação *Wi-Fi*, foi adicionado o ESP-01.

O armazenamento de dados é um ponto que deve ser avaliado numa aplicação com mais de um inversor, pois o espaço de memória do *Raspberry* é bastante limitado, sendo necessário avaliar o período de registro e o quanto de memória se tem disponível.

De modo geral o protótipo cumpriu com seu objetivo de realizar o monitoramento, aquisição e armazenamento dos parâmetros elétricos de um sistema de geração de energia fotovoltaica, gerando histórico e possibilitando vários tipos de implementações, por possuir protocolo aberto. A sua implementação não se restringe apenas para um sistema de geração, mas também pode ser implementado em unidades consumidoras, ou equipamentos específicos que se deseje monitorar, apresentando assim uma grande versatilidade em projetos de medição (seja monofásico ou trifásico).

Como sugestões para futuros trabalhos, recomenda-se:

- Implementação de uma rede de monitoramento da energia gerada pela usina solar Mossoró-II;
- Implementação de técnicas de Inteligência Computacional em conjunto com um medidor de irradiância para uma usina solar fotovoltaica;
- Implementação do sistema de monitoramento numa unidade consumidora e realizar eficiência energética em áreas com alto consumo;
- Implementação de monitoramento de uma carga, implementando Inteligência Computacional para avaliação da performance;
- Implementar o ADE7758 em conjunto com o microcontrolador ESP-32, RTC e módulo cartão SD, possibilitando leitura e armazenamento redundante (sem dependência do servidor);
- Desenvolvimento de biblioteca e aplicação do ADE7754 para medição de parâmetros elétricos em uma rede inteligente (*smart grid*).

## REFERÊNCIAS

ABSOLAR, Associação Brasileira de Energia Solar Fotovoltaica. **Infográfico ABSOLAR**. 2020. ANEEL/ABSOLAR. Disponível em: <http://www.absolar.org.br/infografico-absolar.html>. Acesso em: 16 jun. 2020.

ALIEXPRESS©. **Datasheet TC- SCT013-030**. Disponível em: <https://pt.aliexpress.com/item/32696285299.html>. Acesso em: 20 maio 2020.

AMIN, S. Massoud; WOLLENBERG, B.f.. Toward a smart grid: power delivery for the 21st century. **Ieee Power And Energy Magazine**, [s.l.], v. 3, n. 5, p. 34-41, set. 2005. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/mpae.2005.1507024>.

ANALOG DEVICE©. **Datasheet ADE7758**. Disponível em: <https://www.analog.com/en/products/ade7758.html#product-evaluationkit>. Acesso em: 05 maio 2019.

ANEEL, Agência Nacional de Energia Elétrica. **Resolução Normativa nº 687/2015**: Altera a Resolução Normativa nº 482/2012. Brasil: Aneel, 2015. 25 p. Disponível em: <http://www2.aneel.gov.br/cedoc/ren2015687.pdf>. Acesso em: 07 jun. 2019.

ANEEL, Agência Nacional de Energia Elétrica. **Resolução Normativa nº 733/2016**: Estabelece as condições para a aplicação da modalidade tarifária horária branca. Brasil: Aneel, 2016. 3 p. Disponível em: <https://www2.aneel.gov.br/cedoc/ren2016733.pdf>. Acesso em: 24 jun. 2020.

ANEEL.a, Agência Nacional de Energia Elétrica. **Matriz Elétrica Brasileira**. Referência dos dados 08/06/2020, 13:00h. Microsoft Power BI. Disponível em: <https://app.powerbi.com/view?r=eyJrIjoiNjc4OGYyYjQtYWM2ZC00YjllLWJlYmEtYzdkNTQ1MTc1NjM2IiwidCI6IjQwZDZmOWI4LWVjYTctNDZhMi05MmQ0LWVhNGU5YzAxNzBIMSIsImMiOiR9>. Acesso em: 08 jun. 2020.

ANEEL.b, Agência Nacional de Energia Elétrica. **Unidades Consumidoras com Geração Distribuída**: resumo por tipo de geração. Resumo por Tipo de Geração. 2020. Disponível em: [http://www2.aneel.gov.br/scg/gd/GD\\_Fonte.asp](http://www2.aneel.gov.br/scg/gd/GD_Fonte.asp). Acesso em: 17 jun. 2020.

AUGUSTE. **Biblioteca para ADE7758**. Disponível em: [https://github.com/augustecolle/ADE7758\\_example/blob/master/ADE7758.h](https://github.com/augustecolle/ADE7758_example/blob/master/ADE7758.h). Acesso em: 12 dez. 2019.

BLYNK©. **Blynk server**. 2019. Disponível em: <https://docs.blynk.cc/#blynk-server>. Acesso em: 13 jul. 2019.

BOTTE, B.; CANNATELLI, V.; ROGAI, S.. The telegestore project in ENEL's metering system. **18th International Conference And Exhibition On Electricity Distribution (cired 2005)**, [s.l.], p. 1-4, jun. 2005. IEE. <http://dx.doi.org/10.1049/cp:20051383>.

BRASIL. **Decreto nº 5.163, de 30 de julho de 2004**. Regulamenta a comercialização de energia elétrica, o processo de outorga de concessões e de autorizações de geração de energia elétrica, e dá outras providências. Brasília: Diário Oficial da União, 30 jul. 2004.

CAMARGO, Camila. **Smart Grid: a rede elétrica inteligente**. 2009. Disponível em: <https://www.tecmundo.com.br/internet/3008-smart-grid-a-rede-eletrica-inteligente.htm>. Acesso em: 15 jun. 2020.

CGEE, Centro de Gestão e Estudos Estratégicos. **Redes elétricas inteligentes: contexto nacional**. Brasília: CGEE, 2012. 172 p. (Temas Estratégicos para o Desenvolvimento do Brasil). Disponível em: [https://www.cgee.org.br/documents/10195/734063/Redes\\_Eletricas\\_Inteligentes\\_22mar13\\_9539.pdf/36f87ff1-43ed-4f33-9b53-5c869ace9023?version=1.5](https://www.cgee.org.br/documents/10195/734063/Redes_Eletricas_Inteligentes_22mar13_9539.pdf/36f87ff1-43ed-4f33-9b53-5c869ace9023?version=1.5). Acesso em: 15 jun. 2020.

CRESESB, Centro de Referência para Energia Solar e Eólica de Salvo Brito. **Manual de engenharia para sistemas fotovoltaicos**. Rio de Janeiro, 2014.

DEOE, United States Department of Energy's Office of Electricity. **Smart Grid**. Disponível em: [https://www.smartgrid.gov/the\\_smart\\_grid/smart\\_grid.html](https://www.smartgrid.gov/the_smart_grid/smart_grid.html). Acesso em: 10 jun. 2020.

EPE, Empresa de Pesquisa Energética. **Plano Decenal de Expansão de Energia 2029**. 2020. Disponível em: <http://www.mme.gov.br/documents/36208/461966/01+-+Plano+Decenal+de+Expans%C3%A3o+de+Energia+-+PDE+2029/f7e40439-22c9-da5d-bd4d-1ce68f567ef2>. Acesso em: 08 junho 2020.

FAKIH, Awab *et al.* A Green Energy System for Smart abode. **2018 International Conference On Smart City And Emerging Technology (icscet)**, Mumbai, p.1-5, jan. 2018. IEEE. <http://dx.doi.org/10.1109/icscet.2018.8537365>.

FOREVER, Electronics. **Setup Blynk local server on raspberry pi #iot**. 2019. Disponível em: <https://www.youtube.com/watch?v=sLPbp49LB6g>. Acesso em: 12 jul. 2019.

GÜNDOĞRAN, Cenk; KIETZMANN, Peter; LENDERS, Martine; PETERSEN, Hauke; SCHMIDT, Thomas C.; WÄHLISCH, Matthias. NDN, CoAP, and MQTT. **Proceedings Of The 5Th Acm Conference On Information-Centric Networking**, [S.L.], p. 159-171, 21 set. 2018. ACM. <http://dx.doi.org/10.1145/3267955.3267967>.

HASSAN, Qusay F.. Glossary. **Internet Of Things A To Z**, [s.l.], p. 611-624, 31 maio 2018. John Wiley & Sons, Inc.. <http://dx.doi.org/10.1002/9781119456735.gloss>.

IEEE, The Institute of Electrical and Electronics Engineers. **STD 2413: IEEE Standard for an Architectural Framework for the Internet of Things (IOT)**. New York: Ieee, 2020. 269 p.

INMETRO, Instituto Nacional de Metrologia, Qualidade e Tecnologia. **Portaria Inmetro n.º 587, de 05 de novembro de 2012**. Disponível em: <http://www.inmetro.gov.br/legislacao/rtac/pdf/RTAC001931.pdf>. Acesso em: 31 maio 2020.

IEA, International Energy Agency. **Solar PV – Tracking report june 2020, Paris**. Disponível em: <https://www.iea.org/reports/solar-pv>. Acesso em: 08 junho 2020.

INEE, Instituto Nacional de Eficiência Energética. **O que é geração distribuída**. INEE. Disponível em: [http://www.inee.org.br/forum\\_ger\\_distrib.asp](http://www.inee.org.br/forum_ger_distrib.asp). Acesso em: 16 jun. 2020.

JADHAV, Anurag P.; MALODE, V.b.. Raspberry PI Based OFFLINE MEDIA SERVER. **2019 3rd International Conference On Computing Methodologies And Communication (iccmc)**, [s.l.], p. 531-533, mar. 2019. IEEE. <http://dx.doi.org/10.1109/iccmc.2019.8819718>.

KANTER, Rosabeth Moss; LITOW, Stanley S.. Informed and Interconnected: a manifesto for smarter cities. **Ssrn Electronic Journal**, [s.l.], v. 141, n. 09, p. 1-28, 02 maio 2009. Elsevier BV. <http://dx.doi.org/10.2139/ssrn.1420236>.

KHAIRNAR, Vaibhav C.; K, Kamal Sandeep. Induction Motor Parameter Monitoring System using Zig bee Protocol & MATLAB GUI: automated monitoring system. **2018 Fourth International Conference On Advances In Electrical, Electronics, Information, Communication And Bio-informatics (aeicb)**, [s.l.], p. 1-6, fev. 2018. IEEE. <http://dx.doi.org/10.1109/aeicb.2018.8480992>.

MINIPA© (Santa Catarina). **Proposta técnica - Analisador de energia modelo: ET-5061C**. Disponível em: [http://www.minipa.com.br/images/proposta\\_tecnica/ET-5061C-1303-BR.compressed.pdf](http://www.minipa.com.br/images/proposta_tecnica/ET-5061C-1303-BR.compressed.pdf). Acesso em: 11 jul. 2020.

PATIL, Vinayak Rangrao; PATIL, Manoj D.; KHUDE, Anupam Tanaji. IoT Based Prepaid Energy Meter. **2020 5th International Conference On Devices, Circuits And Systems (ICDCS)**, [s.l.], p. 17-20, mar. 2020. IEEE. <http://dx.doi.org/10.1109/icdcs48716.2020.243539>.

PORTAL SOLAR S.A. (São Paulo) (org.). **Novo medidor de energia da Elsys permite monitoramento simultâneo do consumo e geração de sistemas fotovoltaicos**. 2020. 18/08/2020. Disponível em: <https://www.portalsolar.com.br/blog-solar/energia-renovavel/novo-medidor-de-energia-da-elsys-permite-monitoramento-simultaneo-do-consumo-e-geracao-de-sistemas-fotovoltaicos.html>. Acesso em: 10 set. 2020.

RAMALINGAM, S.; BASKARAN, K.; KALAIARASAN, D.. IoT Enabled Smart Industrial Pollution Monitoring and Control System Using Raspberry Pi with BLYNK Server. **2019 International Conference On Communication And Electronics Systems (ICCES)**, [s.l.], p. 2030-2034, jul. 2019. IEEE. <http://dx.doi.org/10.1109/icces45898.2019.9002430>.

REN21, Renewable Energy Policy Network For The 21st Century. Global status report: Renewables 2020. Paris, 2020. Anual. Disponível em: [https://y3i6c3u4.rocketcdn.me/wp-content/uploads/2019/05/gsr\\_2020\\_full\\_report\\_en.pdf](https://y3i6c3u4.rocketcdn.me/wp-content/uploads/2019/05/gsr_2020_full_report_en.pdf). Acesso em: 16 jun. 2020.

RIGODANZO, Jonas. **Instalação de medidores inteligentes no Brasil: uma análise econômica**. 2015. 118 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal de Santa Maria, Santa Maria, 2015. Disponível em: <https://repositorio.ufsm.br/bitstream/handle/1/8571/RIGODANZO%2c%20JONAS.pdf?sequence=1&isAllowed=y>. Acesso em: 15 jun. 2020.

ROSÁRIO, João Maurício. **Princípios de mecatrônica**. São Paulo: Pearson Prentice Hall, 2005. ISBN: 85-7605-010-2.

SACOMANO, José Benedito; GONÇALVES, Rodrigo Franco; SILVA, Márcia Terra da; BONILLA, Silvia Helena; SÁTYRO, Walter Cardoso; MORAES, Elisângela Mônaco de;

FERIGATTO, Enio Antonio; GLÓRIA JÚNIOR, Irapuan; JACINTHO, José Carlos; REIS, Jacqueline Zonichenn (org.). **Indústria 4.0: Conceitos e fundamentos**. São Paulo: Edgar Blücher Ltda, 2018. ISBN 978-85-212-1371-0.

SAMIE, Farzad; BAUER, Lars; HENKEL, Jörg. IoT technologies for embedded computing. **Proceedings Of The Eleventh Ieee/acm/ifip International Conference On Hardware/software Codesign And System Synthesis - Codes '16**, Pittsburgh, p. 1-10, 2016. ACM Press. <http://dx.doi.org/10.1145/2968456.2974004>.

SANTOS, Bruno P.; SILVA, Lucas A. M.; CELES, Clayson S. F. S.; BORGES NETO, João B.; PERES, Bruna S.; VIEIRA, Marcos Augusto M.; VIEIRA, Luiz Filipe M.; GOUSSEVSKAIA, Olga N.; LOUREIRO, Antonio A. F.. **Internet das Coisas: da Teoria à Prática**. 2016. Universidade Federal de Minas Gerais (UFMG). Disponível em: <https://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-das-coisas.pdf>. Acesso em: 16 jun. 2020.

SARINDA, Jayasinghe L.; IQBAL, Tariq; MANN, George. Low-cost and open source SCADA options for remote control and monitoring of inverters. **2017 Ieee 30Th Canadian Conference On Electrical And Computer Engineering (Ccece)**, Canada, p. 1-4, abr. 2017. IEEE. <http://dx.doi.org/10.1109/ccece.2017.7946658>.

SENEVIRATNE, Pradeeka. **Hands on internet of things with Blynk: build on the power of blynk to configure smart devices and build exciting iot projects**. Birmingham: Packt Publishing, 2018. ISBN 978-1-78899-506-1.

SENAI, Serviço Nacional de Aprendizagem Industrial. **Sistemas lógicos programáveis**. Rio Grande do Sul: Departamento Regional do Rio Grande do Sul, 2013. (Automação Industrial). ISBN 978-85-7519-640-3.

SHINDE, Ashutosh. Challenges in performance monitoring of hyper connected IoT systems. **2016 International Conference On Internet Of Things And Applications (Iota)**, India, p. 1-5, jan. 2016. IEEE. <http://dx.doi.org/10.1109/iota.2016.7562717>.

SMA SOLAR TECHNOLOGY©. **Datasheet Inversor Sunny Boy 3000HF-US**. Disponível em: <http://files.sma.de/dl/9524/SB20-30HF-US-IA-en-12.pdf>. Acesso em: 26 maio 2020.

SMART ENERGY INTERNATIONAL. **Smart grid infrastructure investments in South America to hit \$18.1 billion**. 2020. Disponível em: <https://www.smart-energy.com/industry-sectors/smart-grid/smart-grid-infrastructure-investments-in-south-america-to-hit-18-1-billion/>. Acesso em: 15 jun. 2020.

UFERSA, Universidade Federal Rural do Semi-árido. **Usina Solar Mossoró 01 – Campus Leste: Características gerais da central geradora. Mossoró: UFERSA, 2015**. Disponível em: <https://usinasolar.ufersa.edu.br/wp-content/uploads/sites/130/2019/03/Ficha-Tecnica-USF-CITED.pdf>. Acesso em: 07 jun. 2019.

UPTON, Eben. **Raspberry pi : manual do usuário**. São Paulo: Novatec Editora, 2013. Tradução Celso Roberto Paschoa.

VILLALVA, Marcelo Gradella. **Energia Solar Fotovoltaica Conceitos e Aplicações**. 2. ed. São Paulo: Érica, 2015.

VOLLMER, J.; BERNHARD, L.; SCHIRRMEISTER, C.; ULRICH, M.; VALENZUELA, A.. Flexible machine monitoring by combination of autonomous wireless sensors with a web-interfaced sensor system. **2017 International Conference On Research And Education In Mechatronics (REM)**, [s.l.], p. 1-6, set. 2017. IEEE.  
<http://dx.doi.org/10.1109/rem.2017.8075229>.

XU, Mingwei; WANG, Dongtao. Design of Intelligent Electricity Information Monitoring And Load Forecasting System Based on ADE7758. **2019 Ieee 3rd International Electrical And Energy Conference (CIEEC)**, [s.l.], p. 1399-1403, set. 2019. IEEE.  
<http://dx.doi.org/10.1109/cieec47146.2019.cieec-2019507>.

YE, Feng; QIAN, Yi; HU, Rose Qingyang. Background of the Smart Grid. **Smart Grid Communication Infrastructures**, USA, p. 1-13, 15 jun. 2018. John Wiley & Sons Ltd.  
<http://dx.doi.org/10.1002/9781119240136.ch1>.



## APÊNDICE A

As bibliotecas utilizadas pelo Arduino apresentam geralmente dois tipos de arquivo, sendo eles a extensão .H e .CPP. Segue abaixo a biblioteca utilizada para comunicação do ADE7758 com a IDE do Arduino que teve como base a biblioteca feita segundo Auguste (2019), sendo adicionado a comunicação com os valores de potência ativa, reativa e aparente.

### Arquivo .H

```
#include "Arduino.h"

#define WRITE 0x80 //0x80 = b10000000 voor te schrijven moet BT7 1 zijn

#define AIRMS 0x0A
#define BIRMS 0x0B
#define CIRMS 0x0C
#define AVRMS 0x0D
#define BVRMS 0x0E
#define CVRMS 0x0F
#define AVRMSOS 0x33 //voor calibratie
#define BVRMSOS 0x34
#define CVRMSOS 0x35
#define AIRMSOS 0x36
#define BIRMSOS 0x37
#define CIRMSOS 0x38

#define AWATTHR 0x01
#define BWATTHR 0x02
#define CWATTHR 0x03

#define AVARHR 0x04
#define BVARHR 0x05
#define CVARHR 0x06

#define AVAHR 0x07
#define BVAHR 0x08
#define CVAHR 0x09

#define LCYCMODE 0x17 //The functionalities involved the line-cycle accumulation mode
in the ADE7758 are defined by writing to the LCYCMODE register. Voor de zerocrossings te
tellen moet de ZX per fase hoog gezet worden van dit register (voor alle fases 0x38)
#define MASK 0x18 //When an interrupt event occurs in the ADE7758, the IRQ logic output
goes active low if the mask bit for this event is Logic 1 in the MASK register.

#define STATUS 0x19
#define RSTATUS 0x1A

#define ZXA 0x0200
#define ZXB 0x0400
```

```
#define ZXC 0x0800

#define PHASE_A 0
#define PHASE_B 1
#define PHASE_C 2

class ADE7758{
public:
    ADE7758(int _CS);
    void begin();
    long getVRMS(char phase);
    long getIRMS(char phase);
    long getWATTHR(char phase);
    long getVARHR(char phase);
    long getVAHR(char phase);

    //Later in private te plaatsen
    unsigned char read8bits(char reg);
    unsigned int read16bits(char reg);
    unsigned long read24bits(char reg);

    //write methodes, later ook in private plaatsen
    void write8(char reg, unsigned char data);
    void write16(char reg, unsigned int data);
    void write24(char reg, unsigned long data);

private:
    int CS;

    void enable();
    void disable();
    void calibrateOffset(char phase);
    long getInterruptStatus();
    long getResetInterruptStatus();
};

#endif
```

**Arquivo .CPP**

```

#include "Arduino.h"
#include <SPI.h>
#include "ADE7758.h"
#include <math.h>

ADE7758::ADE7758(int _CS){
    CS = _CS;
}

void ADE7758::begin(){
    pinMode(CS,OUTPUT);
    digitalWrite (CS,HIGH);
    SPI.setDataMode(SPI_MODE2);
    SPI.setClockDivider(SPI_CLOCK_DIV32);
    SPI.setBitOrder(MSBFIRST);
    SPI.begin();
    delay(10);
}

void ADE7758::enable(){
    digitalWrite(CS,LOW);
}

void ADE7758::disable(){
    digitalWrite(CS,HIGH);
}

long ADE7758::getInterruptStatus(){
    return read24bits(STATUS);
}

long ADE7758::getResetInterruptStatus(){
    return read24bits(RSTATUS);
}

unsigned char ADE7758::read8bits(char reg){
    enable();
    unsigned char b0;
    delayMicroseconds(50);
    SPI.transfer(reg);
    delayMicroseconds(50);
    b0=SPI.transfer(0x00);
    delayMicroseconds(50);
    disable();
    return b0;
}

unsigned int ADE7758::read16bits(char reg){
    enable();

```

```

    unsigned char b1,b0;
    delayMicroseconds(50);
    SPI.transfer(reg);
    delayMicroseconds(50);
    b1=SPI.transfer(0x00);
    delayMicroseconds(50);
    b0=SPI.transfer(0x00);
    delayMicroseconds(50);
    disable();
    return (unsigned int)b1<<8 | (unsigned int)b0;
}

unsigned long ADE7758::read24bits(char reg){
    enable();
    unsigned char b2,b1,b0;
    delayMicroseconds(50);
    SPI.transfer(reg);
    delayMicroseconds(50);
    b2=SPI.transfer(0x00);
    delayMicroseconds(50);
    b1=SPI.transfer(0x00);
    delayMicroseconds(50);
    b0=SPI.transfer(0x00);
    delayMicroseconds(50);
    disable();
    return (unsigned long)b2<<16 | (unsigned long)b1<<8 | (unsigned long)b0;
}

void ADE7758::write8(char reg, unsigned char data){
    enable();
    unsigned char data0 = 0;

    // 8th bit (DB7) of the register address controls the Read/Write mode (Refer to spec
page 55 table 13)
    // For Write -> DB7 = 1 / For Read -> DB7 = 0
    reg |= WRITE;
    data0 = (unsigned char)data;

    delayMicroseconds(50);
    SPI.transfer((unsigned char)reg);    //register selection
    delayMicroseconds(50);
    SPI.transfer((unsigned char)data0);
    delayMicroseconds(50);
    disable();
}

void ADE7758::write16(char reg, unsigned int data){
    enable();

```

```

        unsigned char data0=0,data1=0;
        // 8th bit (DB7) of the register address controls the Read/Write mode (Refer to spec
page 55 table 13)
        // For Write -> DB7 = 1 / For Read -> DB7 = 0
        reg |= WRITE;
        //split data
        data0 = (unsigned char)data;
        data1 = (unsigned char)(data>>8);

        //register selection, we have to send a 1 on the 8th bit to perform a write
        delayMicroseconds(50);
        SPI.transfer((unsigned char)reg);
        delayMicroseconds(50);
        //data send, MSB first
        SPI.transfer((unsigned char)data1);
        delayMicroseconds(50);
        SPI.transfer((unsigned char)data0);
        delayMicroseconds(50);
        disable();
    }

void ADE7758::write24(char reg, unsigned long data){
    //METHODE NOG NIET GETEST
    enable();
    unsigned char data0=0,data1=0,data2=0;

    reg |= WRITE;

    data0 = (unsigned char)data;
    data1 = (unsigned char)(data>>8);
    data2 = (unsigned char)(data>>16);
    //float = 32 bit, met de 8 MSBs van data wordt dus geen rekening
    //gehouden
    delayMicroseconds(50);
    SPI.transfer((unsigned char)reg);
    delayMicroseconds(50);
    SPI.transfer((unsigned char)data2);
    delayMicroseconds(50);
    SPI.transfer((unsigned char)data1);
    delayMicroseconds(50);
    SPI.transfer((unsigned char)data0);
    delayMicroseconds(50);
    disable();
}

//To minimize noise synchronize the reading with the zero crossing
long ADE7758::getVRMS(char phase){
    int N = 20; //aantal lezingen voor gemiddelde waarde
    unsigned long VRMS = 0;
    long lastupdate = 0;

```

```

    for (int i = 0; i<N; i++){
        getResetInterruptStatus(); //clear interrupts
        lastupdate = millis();
        while(!(getInterruptStatus() & (ZXA<<phase))){ //Fase afhankelijk gemaakt,
mask shiften met 0,1 of 2
            //Wait for the selected interrupt (zero crossing interrupt)
            if((millis()-lastupdate)>100){
                Serial.println("VRMS Timeout: NaN");
                return -1;
            }
        }
        VRMS += read24bits(AVRMS+phase);
    }
    return VRMS/N; //Fase afhankelijk gemaakt, register AVRMS+0,1 of 2.
}

long ADE7758::getIRMS(char phase){
    int N = 20; //aantal lezingen voor gemiddelde waarde
    unsigned long IRMS = 0;
    long lastupdate = 0;
    for (int i = 0; i<N; i++){
        getResetInterruptStatus(); //clear interrupts
        lastupdate = millis();
        while(!(getInterruptStatus() & (ZXA<<phase))){ //Fase afhankelijk gemaakt,
mask shiften met 0,1 of 2
            //Wait for the selected interrupt (zero crossing interrupt)
            if((millis()-lastupdate)>100){
                Serial.println("IRMS Timeout: NaN");
                return -1;
            }
        }
        IRMS += read24bits(AIRMS+phase);
    }
    return IRMS/N; //Fase afhankelijk gemaakt, register AIRMS+0,1 of 2.
}

void ADE7758::calibrateOffset(char phase){
    //zie datasheet pagina 55
    write8(LCYCMODE, 0x38); //zero crossing voor elke fase
    write24(MASK, 0xE00); //enable de interrupts gegenereerd wanneer er een zero
crossing is

    int Vnom = 400/sqrt(3); //[V] aan te passen per toepassing
    int Vfsc = 260; //[V] full scale spanning, we meten fasespanningen dus 230V +- 23V
+ marge
    int Vmin = Vfsc/20;
    int VRMSmin = 82246; //[-] meting bij Vmin, rauwe data
    int VRMSnom = 1134765; //[-] meting bij Vnom, rauwe data

    int Itest = 10; //[A] aan te passen per calibratie

```

```

int IRMSstest = 0; //[A] meting bij Itest
int IRMSstestmin = 0; //[A] meting bij Itest/500

int Voffset = 1/64*(Vnom*VRMSmin-Vmin*VRMSnom)/(Vmin-Vnom);
int Ioffset = 1/16384*(Itest*Itest*IRMSstest*IRMSstestmin -
(Itest*Itest/250000)*IRMSstestmin*IRMSstestmin)/((1-500*500)*Itest*Itest/250000);

write8(AVRMSOS+phase, Voffset); //registers volgen op elkaar, dus +phase
selecteerd automatisch het juiste register
write8(AIRMSOS+phase, Ioffset);
}

long ADE7758::getWATTHR(char phase){
int N = 20; //aantal lezingen voor gemiddelde waarde
unsigned long WATTHR = 0;
long lastupdate = 0;
for (int i = 0; i<N; i++){
getResetInterruptStatus(); //clear interrupts
lastupdate = millis();
while(!(getInterruptStatus() & (ZXA<<phase))){ //Fase afhankelijk gemaakt,
mask shiften met 0,1 of 2
//Wait for the selected interrupt (zero crossing interrupt)
if((millis()-lastupdate)>100){
Serial.println("WATTHR Timeout: NaN");
return -1;
}
}
WATTHR += read16bits(AWATTHR+phase);
}
return WATTHR/N; //Potencia ativa, register AWATTHR+0,1 of 2.
}

long ADE7758::getVARHR(char phase){
int N = 20; //aantal lezingen voor gemiddelde waarde
unsigned long VARHR = 0;
long lastupdate = 0;
for (int i = 0; i<N; i++){
getResetInterruptStatus(); //clear interrupts
lastupdate = millis();
while(!(getInterruptStatus() & (ZXA<<phase))){ //Fase afhankelijk gemaakt,
mask shiften met 0,1 of 2
//Wait for the selected interrupt (zero crossing interrupt)
if((millis()-lastupdate)>100){
Serial.println("VARHR Timeout: NaN");
return -1;
}
}
VARHR += read16bits(AVARHR+phase);
}
return VARHR/N; //Potencia reativa, register AVARHR+0,1 of 2.
}

```

```

}

long ADE7758::getVAHR(char phase){
  int N = 20; //aantal lezingen voor gemiddelde waarde
  unsigned long VAHR = 0;
  long lastupdate = 0;
  for (int i = 0; i<N; i++){
    getResetInterruptStatus(); //clear interrupts
    lastupdate = millis();
    while(!(getInterruptStatus() & (ZXA<<phase))){ //Fase afhankelijk gemaakt,
mask shiften met 0,1 of 2
      //Wait for the selected interrupt (zero crossing interrupt)
      if((millis()-lastupdate)>100){
        Serial.println("VAHR Timeout: NaN");
        return -1;
      }
    }
    VAHR += read16bits(AVAHR+phase);
  }
  return VAHR/N; //Potencia total, register AVARHR+0,1 of 2.
}

```







Tabela 3: Valor lido pelo Arduino (Tensão A) x valor lido pelo analisador.

<b>Tensão A</b>	<b>Valor Lido (V)</b>
1194242	229,6
1143439	219,9
1094556	210,5
1037842	199,5
912903	175,7
785730	151,2
651964	125,4
518548	99,7
392026	75,3
262703	50,4
133629	25,2
55383	9,2

Fonte: Próprio autor.

Tabela 4: Valor lido pelo Arduino (Corrente A) x valor lido pelo analisador.

<b>Corrente A</b>	<b>Corrente Lida (A)</b>
661	0,0
176275	1,1
359682	2,1
475174	2,7
483729	2,8
543689	3,1
688081	3,9
705042	4,0
958888	5,5
1086066	6,2
1104400	6,3
1316944	7,5
1385096	7,9
1414696	8,1
1627914	9,3
1863909	10,5
1946311	11,1
2067170	11,8

Fonte: Próprio autor.