



CEFET-MG

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
UNIDADE ARAXÁ**

ALEXANDRE LUIZ RIBEIRO

**DESENVOLVIMENTO DE UM SISTEMA DE AQUISIÇÃO DE DADOS E
UM PROGRAMA SUPERVISÓRIO PARA MONITORAMENTO DE
GERAÇÃO DE ENERGIA FOTOVOLTAICA**

ARAXÁ-MG

2017

ALEXANDRE LUIZ RIBEIRO

**DESENVOLVIMENTO DE UM SISTEMA DE AQUISIÇÃO DE DADOS E
UM PROGRAMA SUPERVISÓRIO PARA MONITORAMENTO DE
GERAÇÃO DE ENERGIA FOTOVOLTAICA**

Trabalho de Conclusão de Curso apresentado ao Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá, como requisito parcial para obtenção do grau de Bacharel em Engenharia de Automação Industrial.

Orientador: Prof. Dr. Kleber Lopes Fontoura.

ARAXÁ-MG

2017

ALEXANDRE LUIZ RIBEIRO

**DESENVOLVIMENTO DE UM SISTEMA DE AQUISIÇÃO DE DADOS E
UM PROGRAMA SUPERVISÓRIO PARA MONITORAMENTO DE
GERAÇÃO DE ENERGIA FOTOVOLTAICA**

Trabalho de Conclusão de Curso apresentado ao
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá,
como requisito parcial para obtenção do grau de Bacharel em
Engenharia de Automação Industrial

Defesa: Araxá, 3 de julho de 2017.

BANCA EXAMINADORA

Dr. KLEBER LOPES FONTOURA - Orientador
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

Dr. ADMARÇO VIEIRA DA COSTA - Avaliador
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

Me. LUIS PAULO FAGUNDES - Avaliador
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

DEDICO ESTE TRABALHO

Ao meu orientador, Professor Dr. Kleber Lopes Fontoura e a todos os outros professores que fizeram parte de minha formação acadêmica e que plantaram, de uma forma ou de outra, uma semente do conhecimento que, com certeza, irá prosperar e gerar bons frutos.

AGRADECIMENTOS

À Deus, pelo dom da vida e por me iluminar com a sabedoria e perseverança necessária para conseguir trilhar, de forma honrosa, esse longo e difícil caminho.

À minha linda e digníssima esposa Fernanda, que vivenciou tudo de bom e de ruim dessa trajetória e que, apesar das dificuldades que juntos passamos, sempre esteve ao meu lado me incentivando e dando forças para seguir em frente.

Às minhas duas lindas filhas, Isabella e Giovanna, de quem tanto me ausentei, mas que mesmo assim sempre me deram tanto carinho, sendo fonte de motivação para concluir com sucesso e demonstrar um exemplo de força e dedicação.

Aos meus pais, Domingos e Aparecida, de quem também muito me ausentei, mas que foram de suma importância com suas orações e apoio incondicional.

À minha sogra Magda, que muito me apoiou sempre cuidando de minhas filhas nos momentos em que precisei.

Aos meus irmãos de sangue e irmãos de guerra, que sempre estiveram presentes.

Aos meus companheiros de trabalho que, indiretamente, participaram de alguns momentos.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ), pela oportunidade de participar do projeto de pesquisa e extensão intitulado *Projeto e Instalação de uma Unidade de Microgeração Distribuída baseada na Energia Fotovoltaica com Vistas à Formação de Recursos Humanos e ao Incentivo à Utilização de Sistemas Fotovoltaicos Conectados à Rede Elétrica como Ação de Eficiência Energética*, referente à chamada CNPq - SETEC/MEC Nº 17/2014 - Apoio a Projetos Cooperativos de Pesquisa Aplicada e de Extensão Tecnológica.

A todos os professores, a quem devo o conhecimento adquirido, em especial ao professor Kleber e a professora Jalmira que dedicaram parte extra de seus tempos dando o devido suporte para a elaboração deste trabalho.

Aos servidores da biblioteca, coordenação do curso, registro escolar, laboratório de eletrônica, restaurante e vigilantes, os quais me ajudaram em diversos momentos.

E, a todos os meus companheiros de sala de aula, que vivenciaram juntamente comigo, o que foi cursar uma Engenharia no CEFET.

RESUMO

Essa pesquisa refere-se à elaboração de um Programa Supervisório para monitoramento da geração de energia fotovoltaica por dois sistemas distintos e independentes, compostos por placas solares com características diferentes e também a elaboração de um sistema de aquisição de dados com a função de interfacear a geração de energia com o Programa Supervisório. O principal objetivo desse trabalho foi desenvolver uma aplicação utilizando ferramentas de uso livre e de fácil acesso, como a plataforma micro controlada Arduino e o software para desenvolvimento de aplicativos de monitoramento ScadaBR e, possibilitar a comparação da eficiência energética entre um sistema de geração de energia fotovoltaica utilizando módulos de Silício Monocristalino e outro utilizando módulos de Silício Policristalino.

Palavras chave: Geração de energia fotovoltaica. Programa supervisório. Arduino. ScadaBR. Células de Silício Monocristalino e Policristalino.

ABSTRACT

This research is intended to elaborate a Supervisory Program to monitorate the generation of photovoltaic energy by two distinct and independent systems, composite of solar panels with have different characteristics and also the elaboration of a data acquisition system that will have the function of interfacing the generation with the Supervisory Program. The main objective of this work is to develop an application using free-access tools and easy-to-access such as the Arduino micro-controlled platform and software for the development of ScadaBR monitoring applications and to enable the comparison of energy efficiency between a power generation system Photovoltaic using Monocrystalline Silicon modules and another using Polycrystalline Silicon modules.

Keywords: Generation of photovoltaic energy. Supervisory program. Arduino. ScadaBR. Monocrystalline and Polycrystalline Silicon Cells.

LISTA DE ILUSTRAÇÕES

Figura 1: Especificações técnicas das placas solares. Placa de Silício policristalino a esquerda e placa de Silício monocristalino a direita.	14
Figura 2: Comparativo do GAP entre diferentes tipos de materiais.....	16
Figura 3: Células de silício mono-cristalino.	18
Figura 4: Células de silício poli-cristalino.	19
Figura 5: Sistema fotovoltaico conectado à rede principal.	21
Figura 6: Módulo de Aquisição e Condicionamento de Sinais de Tensão/Corrente. .	22
Figura 7: Aplicação do protocolo MODBUS.	27
Figura 8: Estrutura do protocolo MODBUS.	27
Figura 9: Seleção da placa Arduino Uno no ambiente desenvolvimento Arduino.	30
Figura 10: Caminho de acesso ao exemplo do código SimpleModbusSlave.	31
Figura 11: Código fonte do Programa Supervisório comentado.....	32
Figura 12: Parâmetros a serem configurados no Data Source indicados pelas setas vermelhas.	33
Figura 13: Interface gráfica do Supervisório com menu de ferramentas completo e permissão de administrador.	34
Figura 14: Menu de ferramentas do Supervisório sem permissão de administrador.	35
Figura 15: Watch list.....	35
Figura 16: Janela disponibilizada pelo Supervisório com apresentação do histórico e dados estatísticos de um determinado Data Point.	36
Figura 17: Exemplo de relatório gerado a partir da simulação da variação das entradas analógicas do Arduino.	38
Figura 18: Detalhes do usuário logado.....	39
Figura 19: Alimentação do módulo.....	40
Figura 20: Pinos de medição da tensão 3,3 V.....	40
Figura 21: Calibração do zero.	41
Figura 22: Calibração de offset do canal do sensor de tensão.....	41
Figura 23: Calibração de offset do canal do sensor de corrente.	42
Figura 24: Tensão de 0 Volts aplicada nas entradas analógicas do Arduino.	44
Figura 25: Leitura do sinal de 0 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando a ferramenta Monitor Serial. ...	45

Figura 26: Leitura do sinal de 0 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando o Programa Supervisorio.	46
Figura 27: Parte do código fonte Arduino que trata os valores menores que zero nas entradas analógicas.	46
Figura 28: Tensão de 5 Volts aplicada nas entradas analógicas do Arduino.	47
Figura 29: Leitura do sinal de 5 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando a ferramenta Monitor Serial. ..	47
Figura 30: Leitura do sinal de 5 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando o Programa Supervisorio.	48
Figura 31: Valores de tensão intermediários aplicados nas entradas A3, A2, A1 e A0 do Arduino respectivamente.	48
Figura 32: Leitura das tensões intermediárias aplicadas nas entradas do Arduino por meio da ferramenta Monitor Serial.	49
Figura 33: Leitura das tensões intermediárias aplicadas nas entradas do Arduino por meio do Programa Supervisorio.	49
Figura 34: Parte do código fonte que define os fatores operacionais a serem aplicados no sinal de entrada dos canais do módulo de aquisição de dados.	52
Figura 35: Parte do código fonte que aplica os fatores operacionais nos sinal de entrada dos canais do módulo de aquisição de dados.	54
Figura 36: Leitura do sinal digital do canal de tensão abatendo o zero, utilizando a ferramenta Monitor Serial.	55
Figura 37: Leitura do sinal digital do canal de tensão abatendo o zero, utilizando o Programa Supervisorio.	55
Figura 38: Leitura do sinal do canal de tensão com aplicação dos fatores de conversão, utilizando a ferramenta Monitor Serial.	56
Figura 39: Leitura do sinal do canal de tensão com aplicação dos fatores de conversão, utilizando o Programa Supervisorio.	56
Figura 40: Leitura do sinal digital do canal de corrente abatendo o zero, utilizando a ferramenta Monitor Serial.	57
Figura 41: Leitura do sinal digital do canal de corrente abatendo o zero, utilizando o Programa Supervisorio.	57
Figura 42: Leitura do sinal do canal de corrente com aplicação dos fatores de conversão, utilizando a ferramenta Monitor Serial.	58

Figura 43: Leitura do sinal do canal de corrente com aplicação dos fatores de conversão, utilizando o Programa Supervisório.	58
Figura 44: Especificações técnicas dos módulos fotovoltaicos instalados na micro unidade de geração de energia fotovoltaica nas mediações do laboratório de eletrônica do CEFET - MG / Unidade Araxá.....	59
Figura 45: Relatório de monitoramento de geração de energia fotovoltaica	60

LISTA DE SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
REN	Resolução Normativa
CEFET - MG	Centro Federal de Educação Tecnológica de Minas Gerais
USB	Universal Serial Bus
DSP	Digital Signal Processor
ROM	Memória de Somente Leitura
LED	Diodo Emissor de Luz
IDE	Ambiente de Desenvolvimento Integrado
C++	Linguagem de Programação
I/O	Input / Output
PWM	Modulação Por Largura de Pulso
MHz	Mega Hertz
AC-DC	Corrente Contínua - Corrente Alternada
V	Volts
mA	mili Amper
KB	Kilo Bytes
SRAM	Memória de Acesso Estático Aleatório
EEPROM	Memória de Somente Leitura Programável Apagável Eletricamente
K Ω	Kilo Ohm
UART	Receptor/Transmissor Universal Assíncrono
SCADA	Sistemas de Supervisão e Aquisição de Dados
IHM	Interface Homem Máquina
API	Interface de Programação de Aplicativos
OSI	Open System Interconnection
ADU	Aplication Data Unit
PDU	Protocol Data Unit
RMS	Root Mean Square (Valor eficaz)

SUMÁRIO

1	INTRODUÇÃO	13
2	REFERENCIAL TEÓRICO	16
2.1	Efeito Fotovoltaico	16
2.2	Célula Fotovoltaica	17
2.2.1	CÉLULA DE SILÍCIO MONO-CRISTALINO	17
2.2.2	CÉLULA DE SILÍCIO POLI-CRISTALINO	18
2.3	Módulos Fotovoltaicos	19
2.4	Associação de Módulos Fotovoltaicos	20
2.5	Sistemas Fotovoltaicos Conectados à Rede	20
2.6	Módulo de Aquisição e Condicionamento de Sinais de Tensão e Corrente	21
2.7	Sistemas Embarcados	24
2.7.1	ARDUINO	24
2.7.2	SOFTWARE ARDUINO	25
2.8	ScadaBR	25
2.9	Protocolo de comunicação Modbus	26
3	METODOLOGIA	29
3.1	Configuração do Arduino Software para comunicação com o Arduino Hardware	29
3.2	Criação do código fonte Arduino	30
3.3	Criação do Programa Supervisório no ScadaBR	33
3.4	Configuração do Módulo de Aquisição de Dados	39
4	RESULTADOS E DISCUSSÕES	43
4.1	Testes de hardware, código fonte Arduino e conexão Arduino x Programa Supervisório	43
4.2	Testes do módulo de aquisição de dados	50
4.3	Conexão módulo aquisição de dados x Arduino x Programa Supervisório	52
5	CONCLUSÕES	61
6	TRABALHOS FUTUROS	62
	REFERÊNCIAS	63
	ANEXO A	65

1 INTRODUÇÃO

A geração distribuída é uma produção descentralizada de energia e pode ser feita através de minicentraís geograficamente distribuídas, próximas aos locais de consumo ou até mesmo estabelecidas na própria instalação consumidora, como já ocorre em algumas residências e instituições.

Diferente da geração convencional, ela não necessita de instalação de redes de transmissão, geralmente é obtida por meio de fontes de energia renováveis e pode ser incorporada no sistema elétrico principal. Este tipo de geração oferece uma série de vantagens sobre a geração centralizada tradicional, tais como: redução de perdas elétricas na distribuição, baixa relação custo/benefício, além de causar menores impactos ambientais, por se tratar de uma energia limpa.

As principais fontes renováveis de energia empregadas na geração distribuída são: energia solar, que é obtida por meio de painéis fotovoltaicos, energia eólica, obtida por meio de aerogeradores, e biomassa, obtida por meio de resíduos orgânicos agrícolas e/ou industriais. Pode-se dizer que o sol é a maior fonte de energia dentre essas citadas e seu aproveitamento, tanto como fonte de calor, quanto como fonte de luz é, sem sombra de dúvidas, uma das alternativas energéticas mais promissoras na atualidade.

Diante dessa realidade, o governo brasileiro criou políticas governamentais de incentivo à geração de energia alternativa a partir de fontes renováveis com o objetivo de inserir essa energia na matriz energética principal, conforme descrito no projeto estratégico 013/2011 da Agência Nacional de Energia Elétrica (ANEEL), intitulado *Arranjos Técnicos e Comerciais para Inserção da Geração Solar Fotovoltaica na Matriz Energética Brasileira*, assim como nas resoluções normativas REN 481 e REN 482 (com alterações aprovadas através da RNE 687/15), que regulamentam o processo de compensação de energia por mini e microgeradores de energia alternativa.

Tendo em vista as oportunidades mencionadas, está sendo desenvolvido no Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), na Unidade Araxá, o projeto de pesquisa e extensão intitulado: *Projeto e Instalação de uma Unidade de*

Microgeração Distribuída baseada na Energia Fotovoltaica com Vistas à Formação de Recursos Humanos e ao Incentivo à Utilização de Sistemas Fotovoltaicos Conectados à Rede Elétrica como Ação de Eficiência Energética, sob a coordenação do Prof. Dr. Kleber Lopes Fontoura e Prof. Dr. Admarço Vieira da Costa, que tem como principal objetivo a instalação, nas dependências do CEFET - MG / Unidade Araxá, de uma micro-usina solar fotovoltaica composta de dois sistemas de 3kWp, sendo um composto de 11 placas de Silício monocristalino e outro composto de 11 placas de Silício policristalino, cujas especificações técnicas estão descritas na FIG. 1.

CanadianSolar		CanadianSolar	
MODEL TYPE: CS6P-265P		MODEL TYPE: CS6K-275M	
Nominal Maximum Power (Pmax):	265 W	Nominal Maximum Power (Pmax):	275 W
Optimum Operating Voltage (Vmp):	30.6 V	Optimum Operating Voltage (Vmp):	31.3 V
Optimum Operating Current (Imp):	8.66 A	Optimum Operating Current (Imp):	8.80 A
Open Circuit Voltage (Voc):	37.7 V	Open Circuit Voltage (Voc):	38.3 V
Short Circuit Current (Isc):	9.23 A	Short Circuit Current (Isc):	9.31 A
Maximum System Voltage	: 1000 V	Maximum System Voltage	: 1000 V
Maximum Series Fuse Rating	: 15 A	Maximum Series Fuse Rating	: 15 A

Figura 1: Especificações técnicas das placas solares. Placa de Silício policristalino à esquerda e placa de Silício monocristalino à direita.

Este TCC apresenta-se como um complemento para o projeto de pesquisa e extensão supracitado, propondo-se ao desenvolvimento de um sistema supervisor para monitoramento da energia fotovoltaica gerada pela unidade de microgeração que será implantada, assim como também um sistema para aquisição desses dados.

Para o desenvolvimento do sistema supervisor, será necessário coletar e transmitir, de forma intermitente, a um microcomputador, dados de tensão e corrente elétrica da unidade geradora de energia fotovoltaica para, posteriormente, já de posse dessas grandezas, calcular a potência elétrica que é obtida por meio de uma relação entre elas. Esses dados, por sua vez, serão disponibilizados na tela do computador com representação gráfica e também escalar, de forma a possibilitar o monitoramento contínuo, bem como a análise de desempenho da unidade.

Para coletar os dados da unidade geradora e transmiti-los ao computador será utilizada uma placa de Arduino que é uma plataforma eletrônica utilizada em sistemas embarcados, dotada, dentre outros elementos, de unidade de processamento, memória e de entradas e saídas analógicas e digitais que

possibilitam o interfaceamento entre computador e a unidade geradora. Para a aplicação supracitada, o Arduino terá a função de digitalizar a informação analógica coletada na unidade geradora e enviá-la de forma cabeada, via comunicação Universal Serial Bus (USB), para o computador.

Será utilizado também um módulo de aquisição e condicionamento de sinais de tensão e corrente para fazer a interface entre as placas solares e o Arduino, de modo a ajustar os valores lidos na unidade geradora de energia solar dentro dos níveis especificados no manual do embarcado, bem como fazer o isolamento físico entre eles evitando que o Arduino seja danificado por algum possível surto elétrico. A recepção e o armazenamento dos dados no computador serão gerenciados por um programa supervisorio, que será desenvolvido com a utilização de um software livre e terá a função de disponibilizar as informações na tela para análise e monitoramento do usuário.

Já existem programas que possibilitam o monitoramento e análise de unidades de geração de energia alternativa, porém, são tecnologias pertencentes a fabricantes e, muitas vezes, utilizando ferramentas de hardware e software com elevado custo de aquisição, tornando o projeto ainda mais oneroso. Logo, a implementação de um sistema de monitoramento com a utilização de software e hardware livre, representará significativa redução nos custos da implantação. Outro ponto importante é a possibilidade de se comparar o desempenho e a eficiência energética de dois módulos solares com características construtivas diferentes e consequentemente analisar a viabilidade financeira da implantação de ambos. Esses são fatos que justificam o desenvolvimento desta pesquisa.

Para tanto, tem-se como objetivo geral deste Trabalho de Conclusão de Curso, desenvolver um programa supervisorio para análise e monitoramento de uma unidade de geração de energia fotovoltaica. Visando atingir a este objetivo geral, os objetivos específicos que fundamentaram o desenvolvimento desta pesquisa foram:

- Coletar os dados de uma unidade de microgeração fotovoltaica e transmiti-los via USB para uma central de processamento;
- Monitorar o desempenho de uma unidade de microgeração de energia solar comparando dois tipos de tecnologia de painéis fotovoltaicos;
- Disponibilizar os dados coletados para análise do usuário.

2 REFERENCIAL TEÓRICO

2.1 Efeito Fotovoltaico

Efeito fotovoltaico consiste basicamente na transformação da energia luminosa em energia elétrica. A exposição de um semicondutor a uma fonte de luz faz com que os elétrons que o compõe se excitem ao ponto de criar uma diferença de potencial entre as extremidades do mesmo.

Qualquer elemento químico, seja ele isolante ou condutor elétrico, possui as chamadas bandas de valência e banda de condução. O que distingue cada elemento nesse quesito é a distancia entre tais bandas, a qual é denominada de GAP de energia onde, quanto maior for a distancia entre elas, mais isolante ele é, conforme pode ser visualizado na FIG. 2.

Em baixas temperaturas, a banda de valência estará totalmente preenchida e a banda de condução por sua vez estará totalmente vazia nos semicondutores.

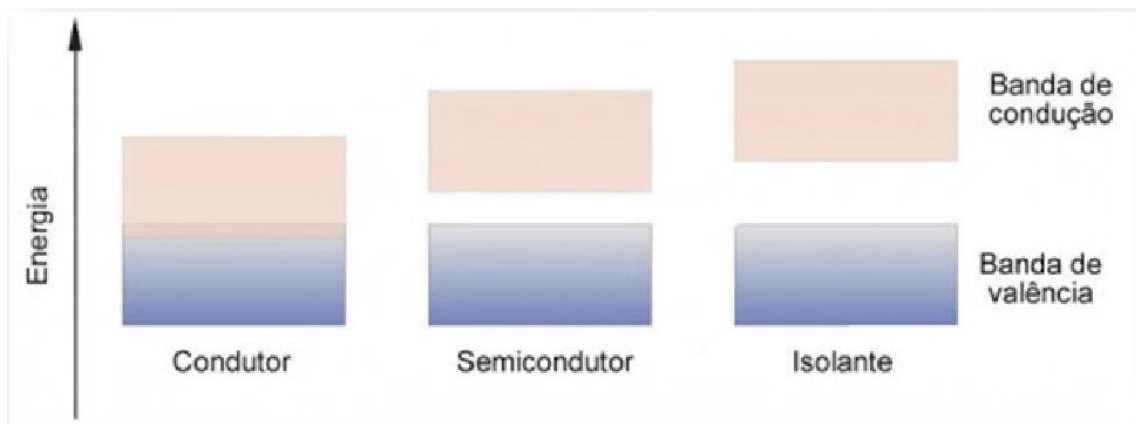


Figura 2: Comparativo do GAP entre diferentes tipos de materiais.
Fonte: SOUSA, R. ([2016?], p. 27).

O silício é o elemento químico mais utilizado para fabricação de semicondutores e, conforme CRESESB (2008, p. 4), *seus átomos se caracterizam por possuírem quatro elétrons que se ligam aos vizinhos, formando uma rede cristalina. Porém, devido à falta de elétrons livres, o silício puro não é um bom condutor. Por este motivo, costuma-se acrescentar, por processo de dopagem, certa porcentagem de fósforo e boro ao mesmo.*

De acordo com CRESESB (2008, p. 4),

Ao serem adicionados átomos com cinco elétrons de ligação, como o fósforo, por exemplo, haverá um elétron em excesso que não poderá ser emparelhado e que ficará "sobrando", fracamente ligado a seu átomo de origem. Isto faz com que, com pouca energia térmica, este elétron se livre, indo para a banda de condução. Diz-se, assim, que o fósforo é um dopante doador de elétrons, denominando-se dopante n ou impureza n.

[...]

Se, por outro lado, forem introduzidos átomos com apenas três elétrons de ligação, como é o caso do boro, haverá falta de um elétron para satisfazer as ligações com os átomos de silício da rede; esta falta de elétron é denominada buraco ou lacuna. Ocorre que, com pouca energia térmica, um elétron de um sítio vizinho pode passar a esta posição, fazendo com que o buraco se desloque. Diz-se, portanto, que o boro é um aceitador de elétrons ou um dopante p.

Sabe-se, portanto que, conforme CRESESB (2008, p. 4), a introdução de átomos de fósforo em uma metade e átomos de boro na outra metade de um material de silício puro, uma junção pn será formada caracterizando um elemento cujas cargas elétricas positivas e negativas se acumulam em extremidades distintas uma da outra. Se as extremidades dessa junção pn forem conectadas externamente por meio de um fio condutor, e a mesma for exposta a uma fonte de luz com energia suficiente para excitar seus elétrons livres ao ponto de romper o gap formado, ocorrerá um deslocamento de carga de uma extremidade à outra, configurando, desta forma o que se chama de efeito fotovoltaico.

2.2 Célula Fotovoltaica

Grande parte das células fotovoltaicas que compõe as placas solares é feitas a partir do silício. O seu uso se deve, além dos fatores físico-químicos já mencionados, a sua abundante existência na face da terra e à larga experiência já adquirida na indústria eletrônica.

Existem vários tipos de células fotovoltaicas a base de silício que são diferenciadas conforme sua estrutura cristalina. As células de silício monocristalinos e de silício poli-cristalinos são as mais utilizadas e serão apresentadas mais adiante.

2.2.1 CÉLULA DE SILÍCIO MONO-CRISTALINO

O rendimento elétrico desse tipo de célula é considerado relativamente elevado, se comparado com a célula de silício poli-cristalino. Em contrapartida, conforme

mencionado em Portal Energia (2011), é um produto que exige complexas e onerosas técnicas de fabricação, com o objetivo de se obter um material com grau de pureza elevado e estrutura cristalina perfeita. É um processo que demanda de mais tempo e consome elevada quantidade de energia.

Para se chegar às qualificações desejadas, o silício é fundido juntamente com uma pequena quantidade de dopante, normalmente o boro, que é do tipo p. Com um fragmento do cristal devidamente orientado e sob rígido controle de temperatura, vai-se extraindo do material fundido um grande cilindro de silício mono cristalino levemente dopado. Este cilindro obtido é cortado em fatias finas de aproximadamente 300 micrômetros.

[...]

Após o corte e limpeza de impurezas das fatias, devem-se introduzir impurezas do tipo N de forma a obter a junção. Este processo é feito através da difusão controlada, em que as fatias de silício são expostas a vapor de fósforo em um forno cuja temperatura varia entre 800 e 1000°C (CRESESB, 2008, p. 4).

A FIG. 3 retrata algumas células fabricadas com Silício monocristalino.

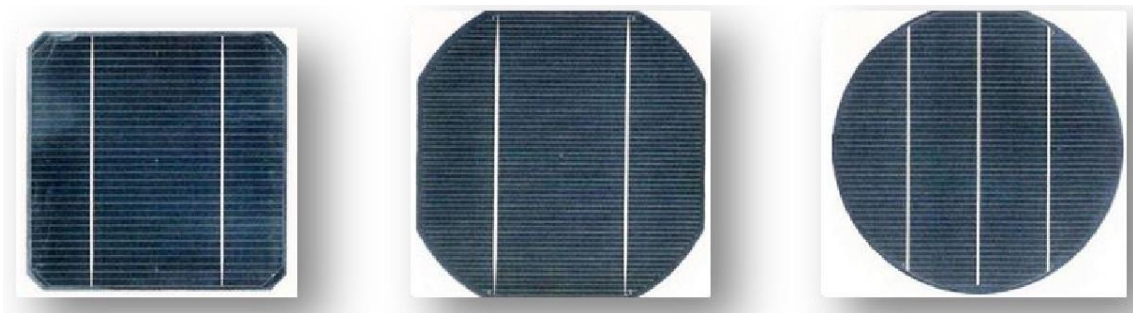


Figura 3: Células de silício mono-cristalino.
Fonte: SOUSA ([2016?], p. 33).

2.2.2 CÉLULA DE SILÍCIO POLI-CRISTALINO

Conforme NIEDZIALKOSKI (2013, p. 7), a quantidade de material utilizado para obtenção das células de silício poli-cristalino é basicamente a mesma para obtenção das células de silício mono-cristalino. A diferença está no processo, o qual consome menos energia, demanda de menos tempo e conseqüentemente de menor custo. Essa simplificação na forma de como elas são fabricadas, resulta em uma célula com estrutura cristalina desorganizada e de menor eficiência se comparada com a mencionada anteriormente.

A FIG. 4 apresenta células fabricadas com Silício policristalino.



Figura 4: Células de silício poli-cristalino.
Fonte: SOUSA ([2016?], p. 34).

2.3 Módulos Fotovoltaicos

De acordo com Sousa ([2016?], p. 40), *uma célula fotovoltaica de silício cristalizado produz tensão de aproximadamente 0,46 a 0,56 volts e uma corrente de aproximadamente 30 mA/cm². As células comerciais geram em torno de 1A, 2,5A, 3A, 5A e 7A.*

Para se obter capacidades de geração de energia suficientes para o consumo, células são conectadas umas nas outras, normalmente em série, formando o que se chama de módulo fotovoltaico. As células que compõe um módulo devem ser compatíveis, ou seja, possuírem características elétricas e de fabricação semelhantes.

A incompatibilidade das células proporciona módulos de baixa qualidade, onde células de menor capacidade limitam o desempenho do conjunto e reduz a eficiência do módulo como um todo.

Cada tipo de módulo, de acordo com a tecnologia utilizada na célula, tem suas características particulares como tipo de célula, dimensão e especificações elétricas como:

- Máxima potência nominal;
- Tensão em máxima potência;
- Tensão em circuito aberto;
- Corrente em máxima potência;
- Corrente de curto circuito.

Um detalhe construtivo é que as células são muito frágeis e necessitam de um encapsulamento em placas rígidas ou flexíveis com o objetivo de prover proteção mecânica e contra as intempéries.

2.4 Associação de Módulos Fotovoltaicos

Um único módulo fotovoltaico não é suficiente para aplicação de geração de energia, sendo necessário ligar eletricamente um conjunto de módulos, a fim de se obter um sistema fotovoltaico que tenha capacidade de fornecer energia suficiente para uma determinada aplicação.

Eles podem ser ligados fazendo associação em série ou em paralelo. Na associação em série, a tensão total do painel é a soma das tensões individuais de cada módulo e a corrente total é a média das correntes de cada módulo. Na associação em paralelo acontece o contrário, ou seja, a corrente total do painel é a soma das correntes individuais de cada painel e a tensão é a média das tensões de cada módulo.

Na maioria dos casos, é necessário associar ligações em série com paralelo em um mesmo sistema fotovoltaico para se obter a especificações de tensão e corrente desejadas no projeto em questão. Não é aconselhável a associação de módulos com capacidades e características distintas, com o intuito de evitar maiores perdas no sistema.

2.5 Sistemas Fotovoltaicos Conectados à Rede

Um sistema fotovoltaico é constituído de uma unidade de geração de energia e uma unidade de condicionamento do sinal gerado.

A unidade de geração é composta por módulos fotovoltaicos que são associados em série ou paralelo (ou em ambos) de acordo com a demanda de energia necessária e a capacidade do equipamento inversor que compõe a unidade de condicionamento do sinal.

A unidade de condicionamento, por sua vez, é composta, conforme citado anteriormente, por um ou mais equipamentos denominados de inversores de

frequência que terão a função de condicionar o sinal gerado de acordo com as especificações elétricas do local de consumo.

A energia gerada pela unidade fotovoltaica tem um sinal contínuo, necessitando, portanto de uma conversão desse sinal alternado de forma a possibilitar a conexão do mesmo ao sistema elétrico principal. Além de realizar essa função, os inversores fazem também o controle dos níveis de tensão, controle e frequência da rede, adequado, de uma forma geral ao sinal do sistema elétrico principal.

Sistemas fotovoltaicos conectados à rede elétrica convencional dispensam a instalação de banco de baterias para armazenamento da energia gerada, visto que ela será consumida diretamente pela carga ou disponibilizada na rede elétrica principal. Esse tipo de sistema se configura como um complemento ao sistema elétrico convencional.

A FIG. 7 retrata, de forma geral, uma unidade de geração de energia fotovoltaica alimentando uma carga qualquer e conectada à rede principal de distribuição de energia ao mesmo tempo.

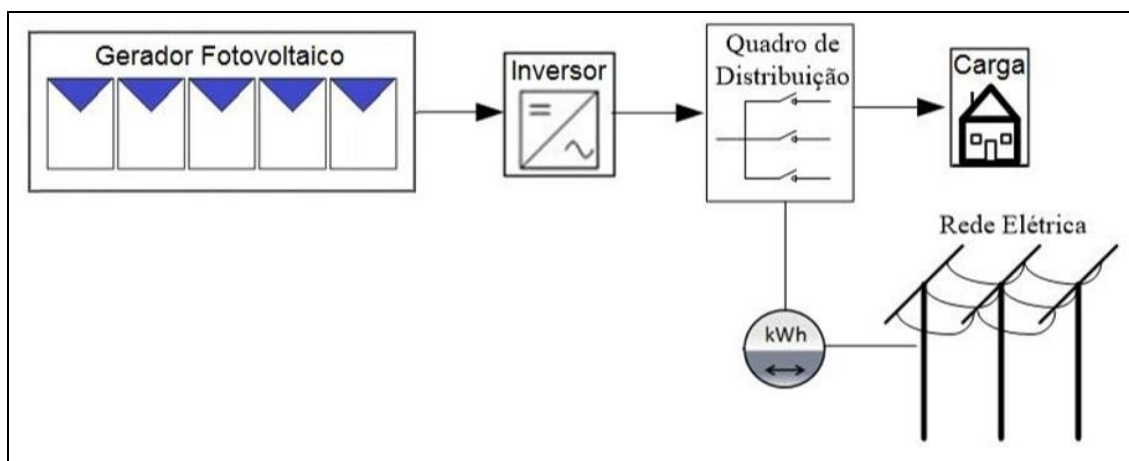


Figura 5: Sistema fotovoltaico conectado à rede principal.
Fonte: PINHO; GALDINO (2014, p. 291).

2.6 Módulo de Aquisição e Condicionamento de Sinais de Tensão e Corrente

O sinal elétrico gerado pelas placas solares possui intensidade elevada, se comparado com intensidade permitida nas portas de entrada de um Arduino.

Por isso, torna-se necessário o acoplamento de um sistema condicionador de sinal entre as placas solares e o Arduino, que adéque o sinal elétrico às condições descritas no manual do Arduino, de forma a garantir a integridade do equipamento.



Figura 6: Módulo de Aquisição e Condicionamento de Sinais de Tensão/Corrente.
Fonte: NUPEP (2011).

O sistema condicionador de sinal, mostrado na FIG. 6, será utilizado para esse fim e apresenta, dentre outras, características de atenuação e isolação física do sinal elétrico.

A atenuação é necessária porque os sinais de tensão e corrente na saída dos sistemas fotovoltaicos propostos são na ordem de 350V e 8 a 9A respectivamente e, portanto, precisam ser reduzidos de modo que seu range de variação se enquadre na faixa de valores permitidos na entrada do Arduino.

A isolação física, por sua vez, tem o objetivo de desacoplar fisicamente o sinal elétrico vindo das placas solares, já que o sistema a ser monitorado pode, ocasionalmente, conter transientes de alta tensão ou curto com terra, fatos que podem danificar o Arduino.

O circuito eletrônico do módulo de aquisição e condicionamento de sinais de tensão e corrente, conforme descrito em NUPEP (2011), é dividido em 3 estágios, sendo eles:

O primeiro estágio tem a função de filtrar e regular a alimentação elétrica do módulo. Fornece um nível contínuo que será considerado como zero pelo Arduino e separa o sinal da tensão com o da corrente de forma a evitar interferência entre os mesmos. Ainda como uma parte anexa a este estágio, há um divisor de tensão, responsável por dar um nível de tensão que, posteriormente, será somado ao sinal com o objetivo de elevar o "zero" da saída (NUPEP, 2011, p. 1).

O segundo estágio é efetivamente responsável pelo tratamento inicial do sinal, sendo que existem dois estágios idênticos: um para o sinal de tensão e outro para o de corrente. Este circuito é responsável por atenuar e limitar o sinal amostrado, devendo este ter seus valores redimensionados de acordo com o interesse do usuário.

O terceiro estágio é, por final, um circuito somador que soma o sinal vindo do estágio anterior a um nível contínuo de aproximadamente 1,5V de forma que o sinal agora possui o nível zero neste valor (NUPEP, 2011, p. 2).

A elevação do zero na saída do módulo de aquisição de sinais é um procedimento necessário e tem o objetivo de atender as especificações técnicas do Digital Signal Processor (DSP) que é um microcontrolador para o qual o respectivo módulo foi criado e não realiza leitura de sinais negativos.

A isolamento física do sinal elétrico gerado pelas placas solares é feita por meio de componentes integrantes do módulo denominados sensores hall onde, o sinal de entrada do módulo alimenta uma bobina, parte de um primeiro circuito, que gera um campo magnético. Este campo, por sua vez, gera uma tensão, que varia proporcionalmente a esse campo, em outro condutor componente de um segundo circuito desconectado fisicamente do primeiro, imerso nesse campo e percorrido por um fluxo de corrente qualquer.

Esse fenômeno é explicado pelo efeito Hall que, conforme Wikipédia (2017, p. 1):

[...] foi descoberto em 1879 por Edwin Hall, que é resultado da força de Lorentz no movimento de elétrons sujeitos a um campo magnético. O efeito Hall determina que, quando se tem um fluxo de corrente em um material sujeito a um campo magnético perpendicular, tem-se então uma tensão nesse mesmo material proporcional ao campo magnético aplicado.

As especificações técnicas do módulo de aquisição e condicionamento de sinais de tensão e corrente estão descritos no Quadro 1 a seguir.

Parâmetro	Min.	Típico	Max.	Unidade
Alimentação +15(relação ao gnd)	14.8	15	15.2	V
Alimentação -15(relação ao gnd)	-14.8	-15	-15.2	V
Corrente Nominal r.m.s medição	-	-	50	A
Faixa Medição de Corrente	-70	-	70	A
Tensão Nominal r.m.s medição*	-	-	500	V
Faixa Medição de Tensão*	-500	-	500	V

Quadro 1: Especificações técnicas do módulo de aquisição e condicionamento de sinais de tensão e corrente.

Fonte: NUPEP (2011).

2.7 Sistemas Embarcados

Conforme EMBEDDED (2017, p. 1), sistema embarcado é um sistema microprocessado com sua unidade de processamento dedicada à realização de tarefas previamente definidas conforme a necessidade da aplicação. Vale-se de recursos computacionais limitados, mas suficientes para execução da atividade que lhe foi confiada.

Um exemplo de sistema embarcado é o Arduino, o qual foi utilizado para desenvolvimento dessa pesquisa.

2.7.1 ARDUINO

Conforme Arduino (2017a):

Arduino é uma plataforma eletrônica, utilizada em sistemas embarcados, de código aberto e baseada em hardware e software fáceis de usar. As placas Arduino são capazes de ler entradas (luz em um sensor, um dedo em um botão ou uma mensagem no Twitter) e transformá-las em uma saída (ativar um motor, ligar um LED - Diodo Emissor de Luz ou publicar algo on-line). Um conjunto de instruções enviadas ao microcontrolador da placa define o que ele irá executar. Para isso, utiliza-se a linguagem de programação Arduino e o Arduino Software Ambiente de desenvolvimento Integrado (IDE).

Além de ser executado em sistemas operacionais Windows e possuir baixo custo de aquisição, de acordo com Arduino (2017a):

Todas as placas Arduino são completamente open-source, capacitando os usuários a construí-los de forma independente e, eventualmente, adaptá-los às suas necessidades específicas. O software também é open-source, possui um ambiente de programação simples e claro, fácil de usar para iniciantes, mas flexível o suficiente para que os usuários avançados possam aproveitar. O idioma pode ser expandido através de bibliotecas C++.

Existem vários tipos de Arduino disponíveis no mercado e, para realização dessa pesquisa, foi adquirido o Arduino UNO e suas especificações técnicas são apresentadas no ANEXO A desse mesmo documento.

2.7.2 SOFTWARE ARDUINO

O Arduino Software (IDE) contém um editor de texto para escrever código, uma área de mensagem, um console de texto, uma barra de ferramentas com botões para funções comuns e uma série de menus. Ele se conecta ao Arduino hardware para fazer o carregamento dos programas criados e se comunicar com eles.

Os programas escritos usando o Arduino Software (IDE) são chamados de esboços. Estes esboços são escritos no editor de texto e são salvos com a extensão de arquivo .ino. O editor possui recursos para cortar/colar e para procurar/substituir texto. A área de mensagem dá feedback enquanto poupando e exportando e também exibe erros. O console exibe a saída de texto pelo software Arduino (IDE), incluindo mensagens de erro completas e outras informações. O canto inferior direito da janela exibe a placa configurada e a porta serial. Os botões da barra de ferramentas permitem verificar e carregar programas, criar, abrir e salvar esboços e abrir o monitor serial (ARDUINO, 2017b).

2.8 ScadaBR

Conforme ScadaBR (2012) Supervisory Control And Data Acquisition (ScadaBR) é *um software de licença gratuita, desenvolvido em modelo "open-source", onde toda a documentação e o código-fonte do sistema estão à disposição, inclusive sendo permitido modificar e redistribuir o software, se necessário.*

Segundo ScadaBR (2012), os pré-requisitos para o correto funcionamento do aplicativo é a execução do Java e um servidor de aplicações (sendo o Apache Tomcat a escolha padrão, incluída em algumas versões) no computador em que ele será utilizado. O ScadaBR possui interface de fácil utilização e oferece visualização das variáveis, gráficos, estatísticas, configuração dos protocolos, alarmes, construção de telas tipo Interface Homem Máquina (IHM) e uma série de opções de configuração.

Sistemas SCADA servem como interface entre o computador e equipamentos eletrônicos como máquinas industriais, controladores automáticos e sensores dos mais variados tipos. Com sistemas SCADA são construídos desde aplicativos simples de sensoriamento e automação, até os famosos *Painéis de Controle* em empresas de geração e distribuição de energia elétrica, centrais de controle de tráfego e assim por diante.

Um SCADA típico oferece drivers de comunicação com equipamentos, um sistema para registro contínuo de dados (datalogger) e uma interface gráfica para usuário onde são disponibilizados elementos gráficos como botões, ícones e displays, representando o processo real que está sendo monitorado ou controlado (SCADABR, 2010).

Dentre as funções que ele executa, as principais são:

- Geração de gráficos com dados históricos de variáveis selecionadas;
- Geração de relatórios com dados estatísticos das variáveis selecionadas;
- Detecção de alarmes e registro de eventos;
- Leitura de parâmetros de variáveis, acionamento e comando de equipamentos.

De acordo com ScadaBR (2010)

[...] após configurar os protocolos de comunicação com os equipamentos e definir as variáveis (entradas e saídas, ou *tags*) de uma aplicação automatizada, é possível montar interfaces de operador Web utilizando o próprio navegador. Também é possível criar aplicativos personalizados, em qualquer linguagem de programação moderna, a partir do código-fonte disponibilizado ou de sua Interface de Programação de Aplicativos (API) *web-services*.

2.9 Protocolo de comunicação Modbus

Modbus é um protocolo aberto e reconhecido no mercado desde 1979, utilizado camada de aplicação do modelo Open System Interconnection (OSI) e fornece condições para estabelecer uma comunicação entre dispositivos cliente/servidor (MODBUS ORGANIZATION, 2012). É um protocolo e pode ser utilizado em diversos tipos de dispositivos de automação, conforme mostrado na FIG. 7.

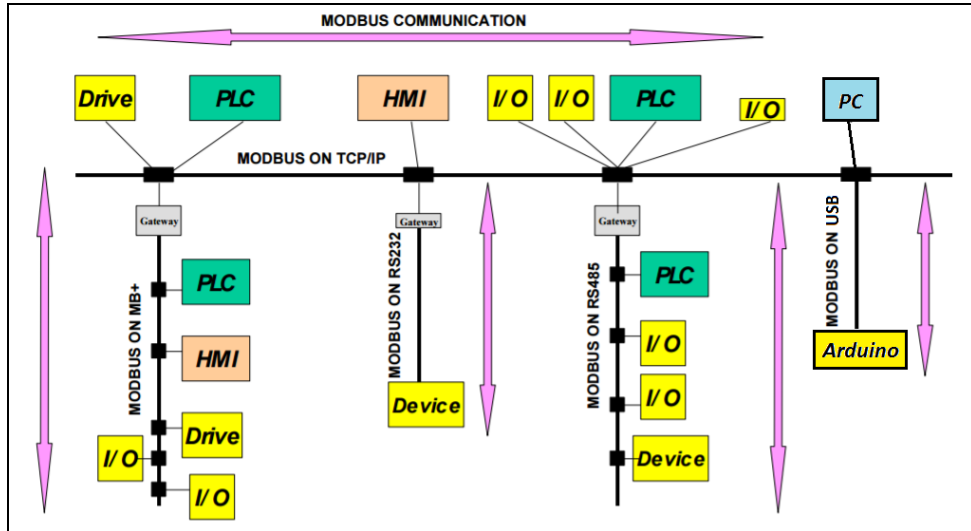


Figura 7: Aplicação do protocolo MODBUS.
 Fonte: Adaptado de MODBUS ORGANIZATION (2012).

O protocolo Modbus define uma unidade simples de dados Protocol Data Unit (PDU) e o endereçamento em cada tipo de rede pode introduzir outros campos, os quais são chamados de unidades de dados de aplicação Application Data Unit (ADU). O pacote PDU é dividido em dois blocos. Em situação normal, ou seja, livre de erros, o primeiro bloco, denominado de Function Code, leva o código da função e o segundo, denominado de Data, leva os dados que serão interpretados pelo servidor conforme codificação enviada em anexo (MODBUS ORGANIZATION, 2012).

Na FIG. 8 pode-se visualizar a representação da estrutura do protocolo MODBUS.

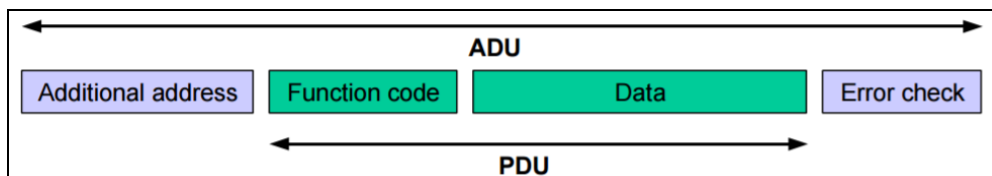


Figura 8: Estrutura do protocolo MODBUS.
 Fonte: MODBUS ORGANIZATION (2012).

Conforme Modbus Organization (2012), *caso ocorra algum tipo de falha durante a transação, o bloco Function Code deixa de levar o código da função para levar o código do erro referente à função original e o respectivo bloco de dados leva a descrição do erro.*

De acordo com Modbus Organization (2012), os códigos das funções são previamente definidos pelo protocolo e seu tamanho é limitado em um byte. O bloco de dados, por sua vez não tem limite de tamanho e pode variar conforme a necessidade da aplicação. Sua única restrição é que somente é enviado um byte de dados por transação.

Segundo Modbus Organization (2012) existem três categorias de códigos de função Modbus, as quais são definidas da seguinte maneira:

- *Códigos de função públicos*: são códigos de funções únicos e bem definidos, de domínio público e garantidos pela organização "Modbus.org".
- *Códigos de função de usuários definidos*: são códigos de funções de domínio público que podem ser selecionados ou implementados pelos próprios usuários, porém não são garantidos pela organização "Modbus.org".
- *Códigos de funções reservados*: não são códigos de domínio público e são utilizados por algumas companhias com produtos específicos.

A função de código 03 (código em hexadecimal) se configura como código de função pública e é denominada de Leitura de Registradores Holding, foi aplicada nesse trabalho para leitura dos registradores holding criados no código fonte Arduino.

Em Modbus Organization (2012) tem-se que a comunicação do protocolo funciona no modelo requisição/resposta, onde o cliente envia uma requisição, no formato PDU, ao servidor que recebe, interpreta e retorna, no mesmo formato, uma resposta conforme a codificação enviada.

De acordo com Modbus Organization (2012), algumas condições são respeitadas pelo protocolo Modbus sobre o meio serial:

- Nenhum servidor (ou escravo) transmite dados sem antes receber uma solicitação do cliente (mestre);
- Somente um mestre de cada vez é conectado à rede e um ou vários escravos (máximo de 247) são conectados na mesma rede;
- Os escravos não se comunicam entre si;
- O mestre abre uma transação Modbus por vêz, podendo ser do tipo Unicast onde o mestre endereça uma mensagem para um único escravo ou Broadcast onde o mestre envia mensagem para todos os escravos ao mesmo tempo (transação sem resposta).

3 METODOLOGIA

Para o desenvolvimento deste trabalho, foram realizadas etapas distintas conforme mostradas no Fluxograma 1.



Fluxograma 1: Etapas de execução do trabalho.

A seguir estão descritos os procedimentos para realização de cada etapa mostrada no Fluxograma 1.

3.1 Configuração do Arduino Software para comunicação com o Arduino Hardware

Existem vários tipos de placas Arduino disponíveis no mercado e todas elas estão relacionadas, para seleção, no ambiente de desenvolvimento Arduino. A placa Arduino utilizada neste trabalho é a Arduino Uno e suas características, bem como suas especificações técnicas já foram apresentadas no referencial teórico mostrado anteriormente. Para salvar corretamente os códigos na placa Arduino Uno, é necessário selecioná-la no ambiente de desenvolvimento Arduino e para isto, basta ir ao menu *Ferramentas* e depois *Placa*, conforme mostra a FIG. 9.

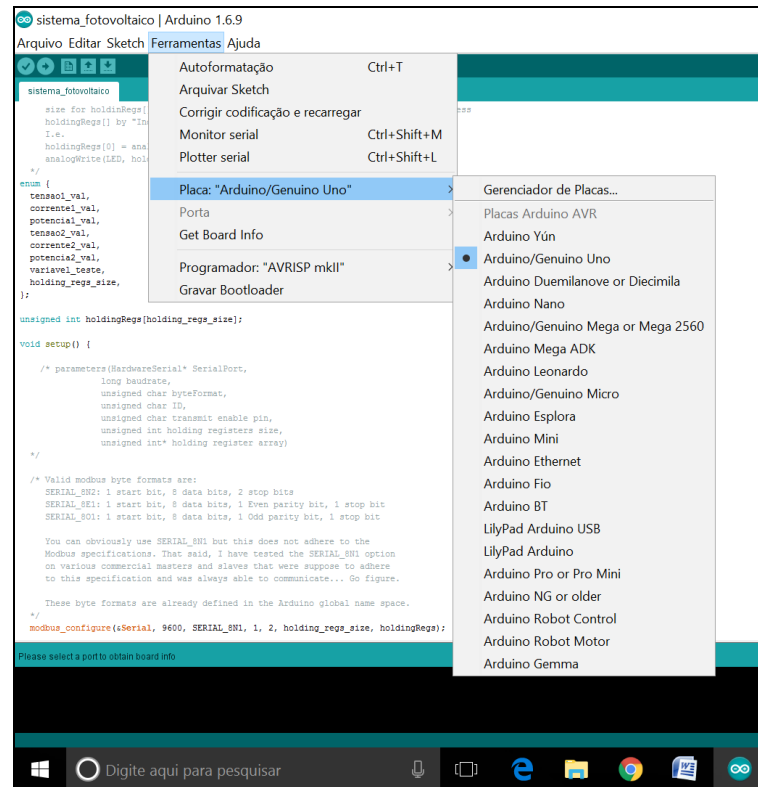


Figura 9: Seleção da placa Arduino Uno no ambiente desenvolvimento Arduino.

3.2 Criação do código fonte Arduino

Para que a comunicação serial entre a placa Arduino e o Sistema ScadaBR seja possível, é necessária a utilização de um protocolo de comunicação denominado *Modbus*, o qual não é disponibilizado na versão padrão do Arduino Software. Para tanto é necessário, portanto, obter, por meio da internet, a biblioteca denominada *SimpleModbusSlave* que fornece condições para manipulação de tal protocolo no ambiente de desenvolvimento Arduino. Essa biblioteca foi copiada para dentro de uma pasta criada com a seguinte descrição: *SimpleModbusSlave*, localizada no seguinte caminho: *C:\Program Files (x86)\Arduino\libraries*.

Visando facilitar a vida do usuário, as bibliotecas instaladas no Arduino Software disponibilizam exemplos de códigos, os quais são utilizados como escopo e podem ser manipulados conforme a necessidade da aplicação.

Para acessar o exemplo de código do *SimpleModbusSlave* basta ir ao menu *Arquivo*, depois *Exemplos*, em seguida *SimpleModbusSlaveV10* e depois *SimpleModbusSlaveArduino*, conforme mostrado na FIG. 10. Uma nova janela é aberta contendo o escopo do respectivo código Arduino.

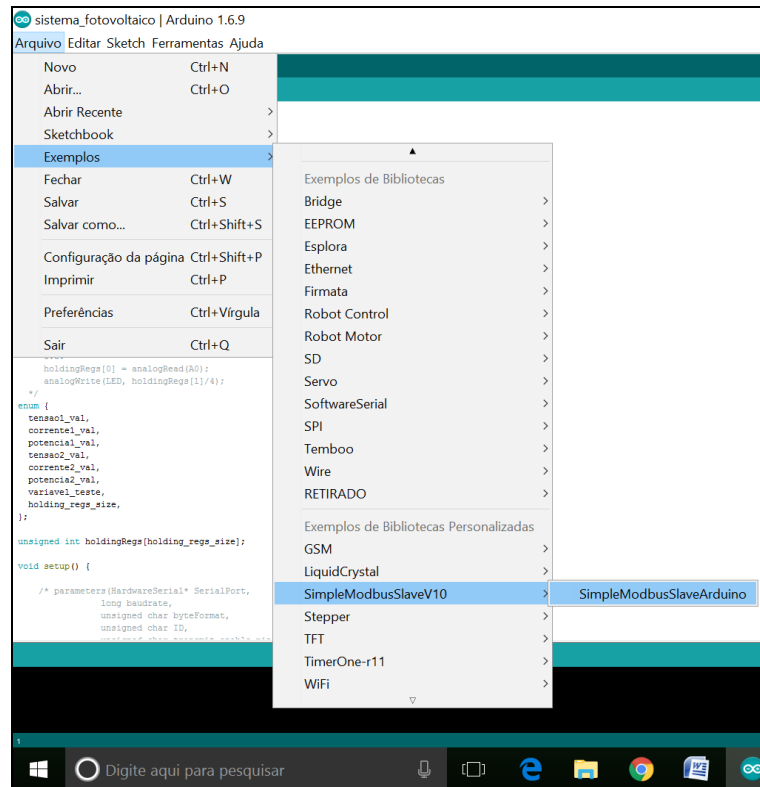


Figura 10: Caminho de acesso ao exemplo do código SimpleModbusSlave.

O código manipulado neste momento pode ser visualizado na FIG. 11 e é caracterizado, de uma forma geral, por nomear e definir como entrada as portas analógicas utilizadas na placa Arduino, por configurar o protocolo *modbus* e por carregar os dados lidos nas portas analógicas em variáveis para envio ao Supervisório.

Nesse mesmo código, foram criadas também algumas variáveis que receberão sinais das entradas analógicas do Arduino para, quando necessário, serem monitoradas por meio da ferramenta *Monitor serial*, a qual auxilia no envio e recebimento de informações para a placa Arduino sem a necessidade de recorrer a uma ferramenta externa.

```

sistema_fotovoltaico
// Inclusão da biblioteca necessária para estabelecer comunicação entre Arduino UNO e ScadaBR
#include <SimpleModbusSlave.h>

// Nomeação das portas analógicas do Arduino utilizadas
#define tensao_mono A0
#define corrente_mono A1
#define tensao_poli A2
#define corrente_poli A3

// Declaração de variáveis para teste do código na ferramenta Monitor serial
double tensao_mono_analogica = 0;
double corrente_mono_analogica = 0;
double tensao_poli_analogica = 0;
double corrente_poli_analogica = 0;

// Declaração das variáveis fator, a qual ...
// ... do sinal amostrado nas entradas analógicas A0 e A2
double zero_tensao = 0; // Variável que irá armazenar o valor do zero setado no canal de tensão da placa de aquisição de dados
double zero_corrente = 0; // Variável que irá armazenar o valor do zero setado no canal de corrente da placa de aquisição de dados
double fator_tensao = 0; // Variável que irá armazenar o fator de conversão binário/decimal do sinal de tensão
double fator_corrente = 0; // Variável que irá armazenar o fator de conversão binário/decimal do sinal de corrente

enum {
// Declaração de registradores que armazenarão os dados lidos nas portas analógicas (holdingRegs)
tensao_mono_val,
corrente_mono_val,
potencia_mono_val,
tensao_poli_val,
corrente_poli_val,
potencia_poli_val,
// Declaração do registrador que contém a informação utilizada pela biblioteca para identificar ...
// ... a quantidade de Registradores Holding que estão sendo utilizados
holding_regs_size,
};

// Declaração da variável de manipulação dos registradores holdingRegs
unsigned int holdingRegs[holding_regs_size];

void setup() {

// parâmetros de configuração do modbus:
// * Serial = definição da porta utilizada;
// * 9600 = taxa de transmissão dos dados;
// * SERIAL_8N1 = formato do pacote utilizado no protocolo
// formatos válidos:
// SERIAL_8N2: 1 start bit, 8 data bits, 2 stop bits
// SERIAL_8E1: 1 start bit, 8 data bits, 1 Even parity bit, 1 stop bit
// SERIAL_8O1: 1 start bit, 8 data bits, 1 Odd parity bit, 1 stop bit
// * 1 = identificação do escravo;
// * 2 = pino de transmissão habilitado;
// * holding_regs_size = tamanho do registrador holdingRegs;
// * holdingRegs = registrador de armazenamento dos dados das portas analógicas */
modbus_configure(&Serial, 9600, SERIAL_8N1, 1, 2, holding_regs_size, holdingRegs);

// Atualização dos principais parâmetros de configuração do protocolo
modbus_update_comms(9600, SERIAL_8N1, 1);

pinMode(tensao_mono, INPUT); // define pino "tensao_painel_1" como entrada
pinMode(corrente_mono, INPUT); // define pino "corrente_painel_1" como entrada
pinMode(tensao_poli, INPUT); // define pino "tensao_painel_2" como entrada
pinMode(corrente_poli, INPUT); // define pino "corrente_painel_2" como entrada

// Parâmetro de configuração da taxa de transmissão da ferramenta Monitor serial
Serial.begin(9600);

// Menu do Monitor serial
Serial.println("\ntensao_mono\t\tcorrente_mono\t\ttensao_poli\t\tcorrente_poli\n");

// Definição dos valores do zero e fatores de conversão
zero_tensao = 1023 / 5 * 1.5; // Valor em binário
zero_corrente = 1023 / 5 * 1.514; // Valor em binário
fator_tensao = 350 / (1023 / 5 * (2.02 - 1.5)); // Valor em decimal
fator_corrente = 9 / (1023 / 5 * (3.404 - 1.514)); // Valor em decimal
}

void loop() {

// Atualização periódica do protocolo de comunicação modbus
modbus_update();

// Ajuste dos dados lidos nas entradas analógicas e direcionamento para as variáveis declaradas
tensao_mono_analogica = (analogRead(tensao_mono) - zero_tensao) * fator_tensao;
corrente_mono_analogica = (analogRead(corrente_mono) - zero_corrente) * fator_corrente * 1000;
tensao_poli_analogica = (analogRead(tensao_poli) - zero_tensao) * fator_tensao;
corrente_poli_analogica = (analogRead(corrente_poli) - zero_corrente) * fator_corrente * 1000;

// Tratamento para fundo de escala inferior da leitura das portas analógicas não serem menor que zero
if (tensao_mono_analogica < 0)
tensao_mono_analogica = 0;
if (corrente_mono_analogica < 0)
corrente_mono_analogica = 0;
if (tensao_poli_analogica < 0)
tensao_poli_analogica = 0;
if (corrente_poli_analogica < 0)
corrente_poli_analogica = 0;

// Carregamento dos dados das variáveis declaradas nos registradores holdingRegs
holdingRegs[tensao_mono_val] = tensao_mono_analogica;
holdingRegs[corrente_mono_val] = corrente_mono_analogica;
holdingRegs[potencia_mono_val] = tensao_mono_analogica * corrente_mono_analogica;
holdingRegs[tensao_poli_val] = tensao_poli_analogica;
holdingRegs[corrente_poli_val] = corrente_poli_analogica;
holdingRegs[potencia_poli_val] = tensao_poli_analogica * corrente_poli_analogica;

// Impressão dos dados das variáveis de teste na ferramenta Monitor serial
Serial.print(tensao_mono_analogica);
Serial.print("\t\t");
Serial.print(corrente_mono_analogica);
Serial.print("\t\t");
Serial.print(tensao_poli_analogica);
Serial.print("\t\t");
Serial.println(corrente_poli_analogica);
}

```

Figura 11: Código fonte do Programa Supervisório comentado.

3.3 Criação do Programa Supervisório no ScadaBR

O primeiro passo para se desenvolver o Programa Supervisório é a criação de um banco de dados, denominado pelo sistema *ScadaBR* de Data Source, com suas respectivas variáveis locais, chamadas de *Data Points*, as quais receberão os dados enviados pelo *Arduino*.

O data source deverá ser do tipo *Modbus Serial* e para cada variável, manipulada pelo registrador, denominado *holdingRegs*, no código fonte *Arduino*, é necessária a criação de uma variável no banco de dados *ScadaBR* para recebimento dos respectivos dados. Na FIG. 12 os parâmetros configurados no data source são indicados com seta vermelha.

The screenshot shows the ScadaBR 1.0 CE web interface for configuring a Data Source. The main configuration area is titled 'Propriedades do modbus serial' and includes the following fields:

- Nome:** SISTEMA_FOTOVOLTAI
- Export ID (XID):** DS_106318
- Período de atualização:** 1 segundo(s)
- Quantificação:**
- Timeout (ms):** 500
- Tentativas:** 3
- Apenas quantidades contínuas:**
- Criar pontos de monitor de escravo:**
- Máxima contagem de leitura de bits:** 2000
- Máxima contagem de leitura de registradores:** 125
- Máxima contagem de escrita de registradores:** 120
- Porta:** COM3
- Baud rate:** 9600
- Controle de fluxo de entrada:** Nenhum
- Controle de fluxo de saída:** Nenhum
- Data bits:** 8
- Stop bits:** 1
- Parity:** Nenhuma
- Codificação:** RTU
- Echo:** Desligado
- Simultaneidade:** Função

Other sections include 'Pesquisa de nós modbus' (empty), 'Leitura de dados modbus' (Id do escravo: 1, Faixa do registro: Status do coil, Offset: 0, Registradores: 100), and 'Teste de localizador de ponto' (Id do escravo: 1, Faixa do registro: Status do coil, Tipo de dados modbus: Binário, Bit: 0, Registradores: 0, Codificação: ASCII).

The 'Data points' table at the bottom is as follows:

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
corrente_painel1	Numérico		1	Registrador holding 1	
corrente_painel2	Numérico		1	Registrador holding 4	
potencia_painel1	Numérico		1	Registrador holding 2	
potencia_painel2	Numérico		1	Registrador holding 5	
tensao_painel1	Numérico		1	Registrador holding 0	
tensao_painel2	Numérico		1	Registrador holding 3	
variavel_teste	Numérico		1	Registrador holding 6	

Figura 12: Parâmetros a serem configurados no Data Source indicados pelas setas vermelhas.

Com o *Data Source* criado e configurado, uma interface gráfica para visualização e supervisão dos dados se torna necessária visando facilitar a interação com o usuário. Para criação dessa interface, foi utilizado o recurso de *Representa gráfica*, disponibilizada no menu de ferramentas do ScadaBR.

Uma imagem, que retrata o sistema fotovoltaico como um todo, foi criada para utilização como plano de fundo na tela de supervisão e os indicadores de tensão, corrente e potência das placas de Silício Monocristalino, assim como os mesmos indicadores das placas de Silício Policristalinos foram disponibilizados, sobrepondo a respectiva imagem, como mostrado na FIG. 13. Também foi disponibilizado na tela de supervisão um gráfico que relaciona as potências dos dois conjuntos de placas em função do tempo, com o objetivo de comparar a eficiência energética entre ambos. Os dados dos respectivos indicadores são plotados na tela do Supervisório em tempo real.

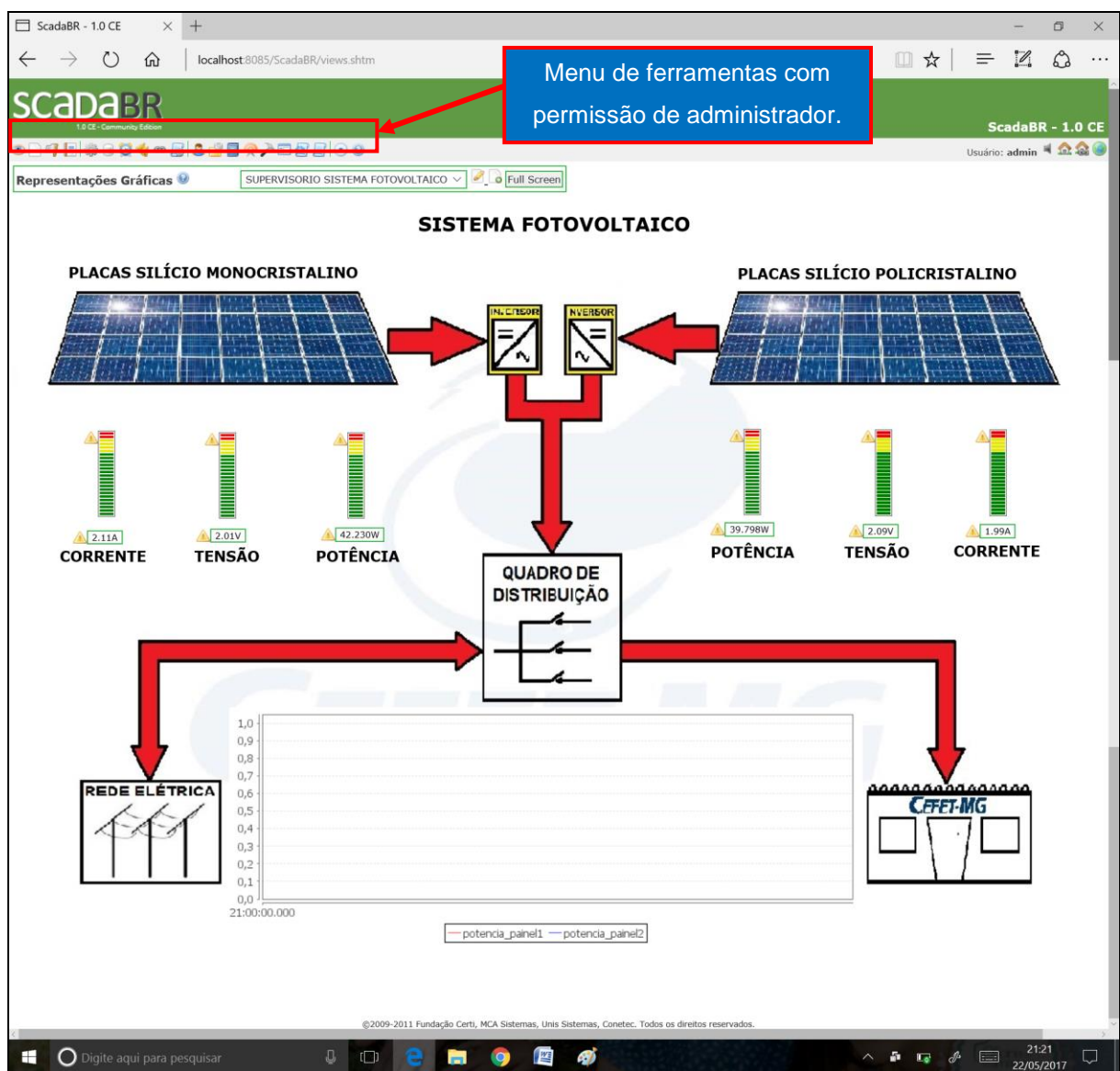


Figura 13: Interface gráfica do Supervisório com menu de ferramentas completo e permissão de administrador.

Para desenvolver o Sistema Supervisório foi necessário logar no ScadaBR com permissão de administrador, a qual possibilita acesso a todas as ferramentas disponíveis na versão instalada no computador, além de fornecer condições de criação, edição e exclusão em qualquer aplicação manipulada no software em questão.

Porém, visando impossibilitar o acesso às ferramentas mencionadas anteriormente, com vistas a preservar o correto funcionamento do Supervisório, foi criada uma nova permissão de acesso, limitando o usuário final aos recursos considerados suficientes para o monitoramento da geração de energia fotovoltaica, conforme FIG. 14.



Figura 14: Menu de ferramentas do Supervisório sem permissão de administrador.

A seguir estão descritas as funções de cada ícone disponibilizado no menu de ferramentas mostrado na FIG. 14.

O primeiro item do menu é denominado *Watch list*. Ele disponibiliza todos os *Data Points* para que uma lista de visualização contendo apenas as informações de interesse do usuário possa ser criada. Nessa opção os dados também são atualizados em tempo real assim como na representação gráfica.

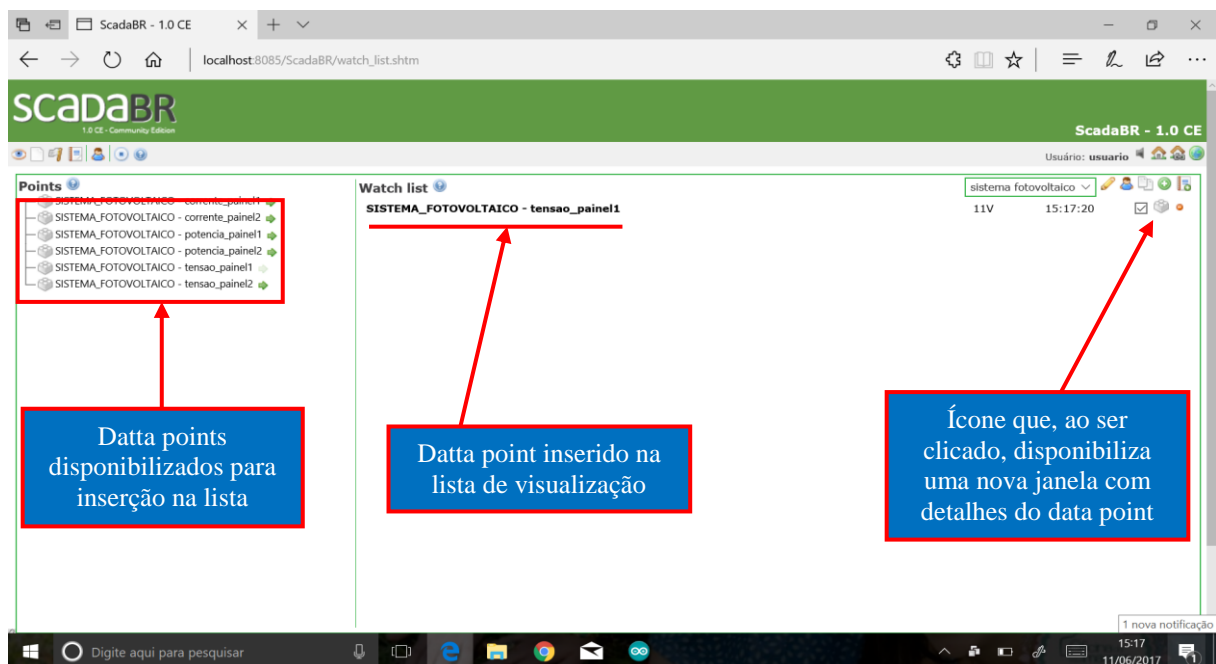


Figura 15: Watch list.

Para cada *Data Point* adicionado na *Watch list* criada pelo usuário, existe um ícone localizado na extremidade direita da tela, conforme mostrado na FIG. 15, que ao ser clicado, carrega uma nova janela contendo todos os detalhes do respectivo *Data Point*, bem como um gráfico com opção de demonstração do histórico de períodos anteriores. Vide FIG. 16.

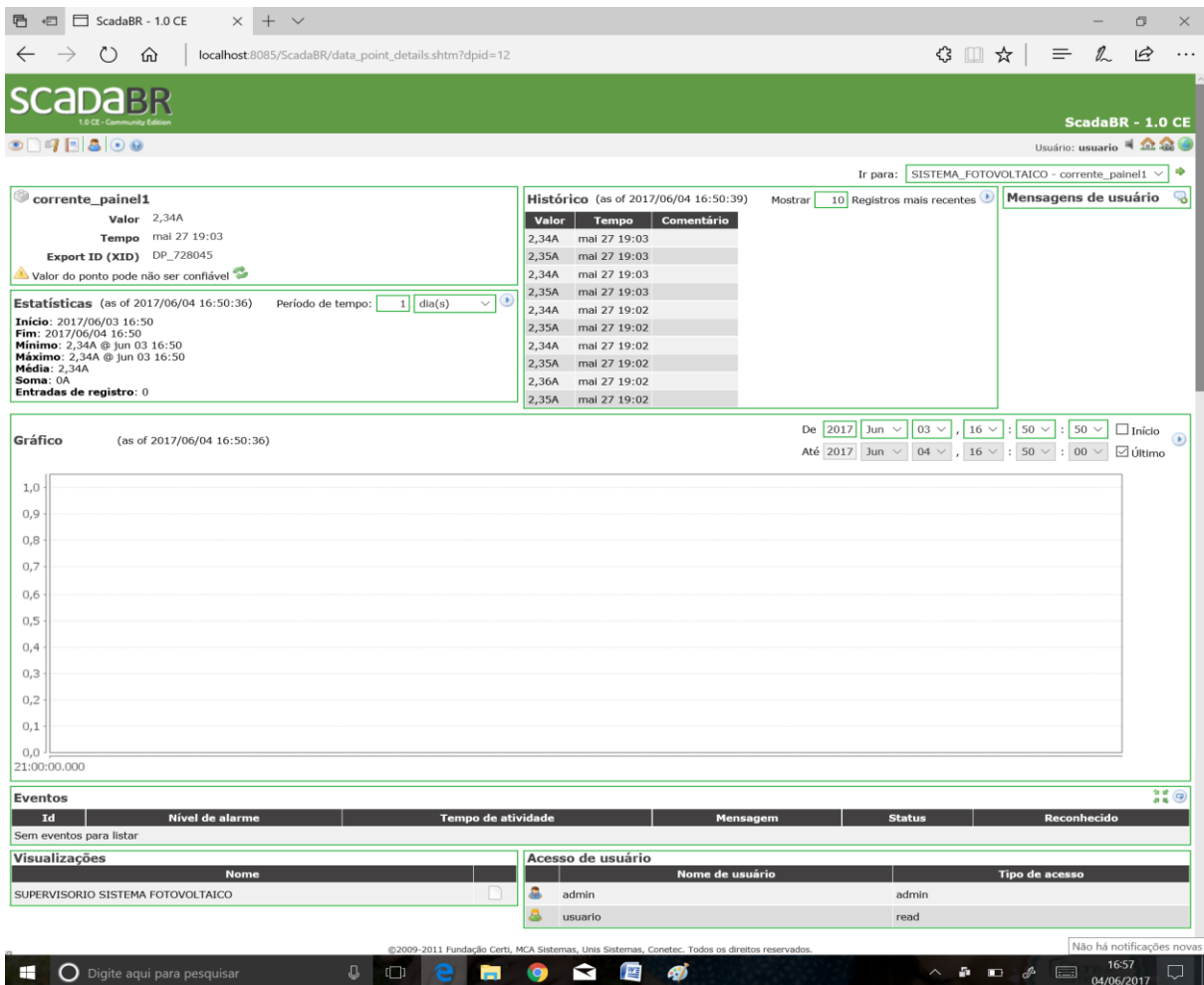


Figura 16: Janela disponibilizada pelo Supervisório com apresentação do histórico e dados estatísticos de um determinado Data Point.

O segundo ícone do menu, ao ser clicado, disponibiliza a representação gráfica do Supervisório criado, onde todos os *Data Points* podem ser visualizados ao mesmo tempo. Todos os indicadores são mostrados em uma única tela contendo representações simbólicas dos dois sistemas fotovoltaicos e seus respectivos inversores de frequência, da energia gerada sendo alimentada no quadro de distribuição, da rede elétrica, da unidade CEFET-MG e do sentido do fluxo da energia gerada, conforme FIG. 13 mostrada anteriormente.

O terceiro ícone do menu disponibiliza uma relação de ocorrência de alarmes ocorridos durante monitoramento do sistema de geração de energia fotovoltaica que, no entanto, não é contemplada a configuração desse item nesse trabalho.

O quarto ícone disponibiliza uma ferramenta para criação de relatórios, a qual fornece opções de geração periódica ou em datas específicas, podendo ser inserido quaisquer que sejam os *Data Points* de interesse do usuário. Um exemplo de relatório gerado nessa ferramenta é apresentado na FIG. 17, onde pode ser observado uma evolução gradativa de quatro sinais, em escala digital (0 a 1023), obtida por meio da aplicação de uma tensão nas entradas analógicas do Arduino variando de 0 a 5 Volts.

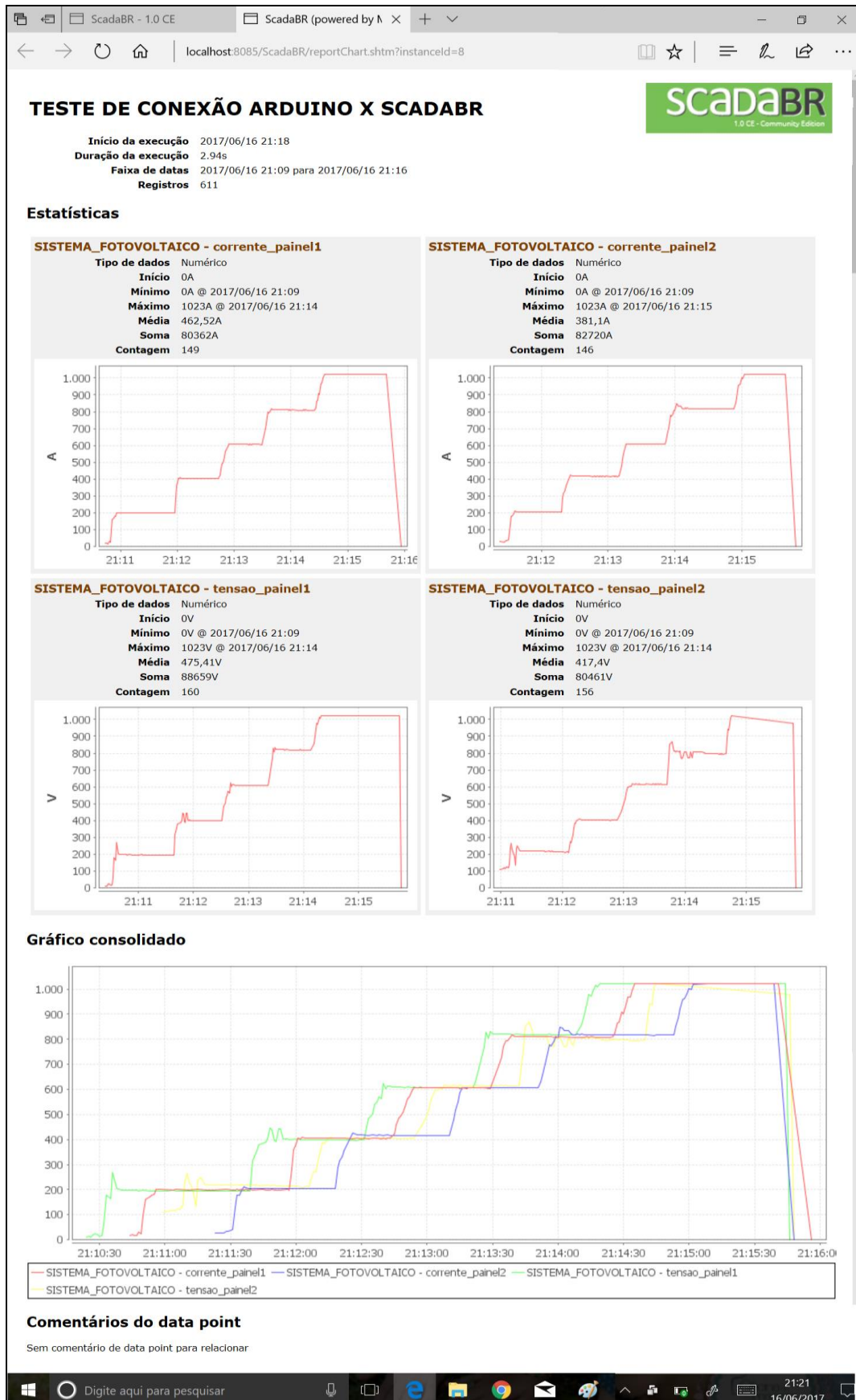
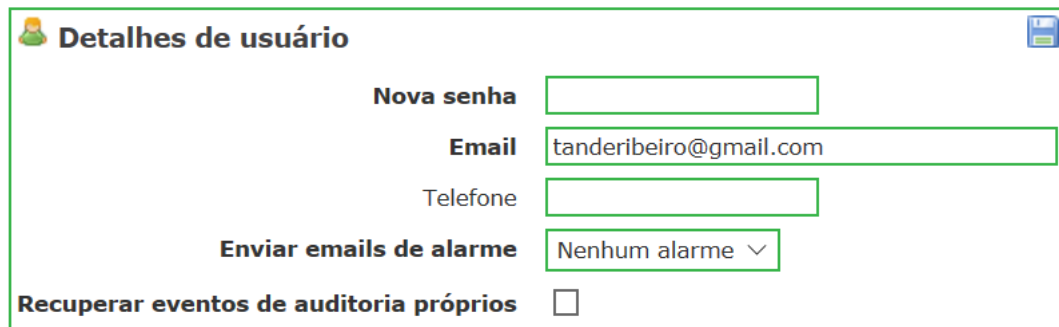


Figura 17: Exemplo de relatório gerado a partir da simulação da variação das entradas analógicas do Arduino.

O quinto ícone do menu, ao ser clicado, disponibiliza uma janela, mostrada na FIG. 18, contendo informações pessoais do usuário logado que podem ser simplesmente visualizadas ou alteradas conforme necessidade. Os campos disponíveis são inserção de nova senha telefone e e-mail. Esse trabalho não contempla o tratamento de envio de notificações por e-mail.



A imagem mostra uma janela de interface de usuário intitulada "Detalhes de usuário". No topo à esquerda há um ícone de perfil de usuário e o título "Detalhes de usuário". No canto superior direito há um ícone de fechamento. O formulário contém os seguintes elementos:

- Um campo de texto rotulado "Nova senha".
- Um campo de texto rotulado "Email" com o valor "tanderibeiro@gmail.com" preenchido.
- Um campo de texto rotulado "Telefone".
- Um menu suspenso rotulado "Enviar emails de alarme" com a opção selecionada "Nenhum alarme".
- Um campo de texto rotulado "Recuperar eventos de auditoria próprios" com uma caixa de seleção vazia.

Figura 18: Detalhes do usuário logado.

O acionamento do sexto ícone é o caminho para realização do logout do usuário no Sistema Supervisório e, para finalizar, o sétimo é o ícone de ajuda que, ao ser clicado, disponibiliza uma janela relacionando uma apresentação geral de como a aplicação funciona além de conceitos que os usuários possam vir a necessitar durante a utilização.

3.4 Configuração do Módulo de Aquisição de Dados

Nessa etapa do trabalho, foi realizada a calibração e o ajuste do módulo de aquisição e condicionamento de sinais de tensão e corrente conforme recomendação do fabricante. Esse dispositivo foi disponibilizado pelo CEFET-MG Unidade Araxá.

O referido módulo possui 3 potenciômetros de calibração, sendo dois para regular o offset dos sinais de saída e um para regular a tensão de referência "Vref", que será o meio lógico interpretado pelo Arduino.

Para realizar a calibração do módulo, foram executados os seguintes passos:

1. Primeiramente foi definido o valor do resistor de carga da entrada do sensor de tensão, de forma que, após a queda de tensão no mesmo, o valor RMS da corrente

que alimenta o sensor seja de, no máximo, 10 mA. Considerando a tensão vinda das placas solares de, no máximo, 350 V e aplicando a lei de ohms, o resistor será de:

$$R = \frac{V}{I} = \frac{350}{0,01} = 35Kohms$$

Como não havia disponível um resistor de 35 Kohms, foi utilizado um resistor de 50 Kohms que, mesmo possuindo maior resistência, pode ser aplicado, pois apesar de limitar a corrente de alimentação do sensor em 7 mA, não ultrapassa o valor limite de previamente especificado de 10 mA.

2. A placa foi alimentada com fonte externa simétrica, conectando os cabos de +15V, GND e -15V nos respectivos bornes indicados na FIG. 19.

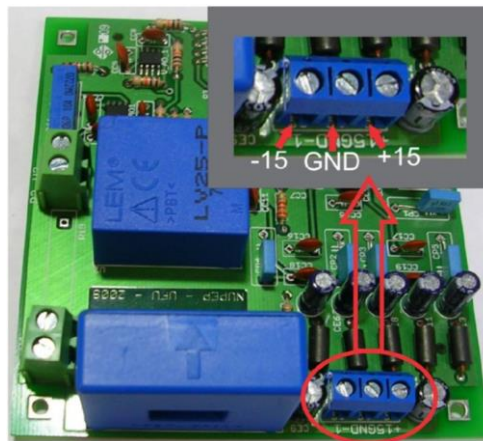


Figura 19: Alimentação do módulo.
Fonte: NUPEP (2011).

3. No pino apresentado na FIG. 20, foi conferida a presença de uma tensão estabilizada em 3.3V em relação ao GND. O valor medido foi de 3,27 V.

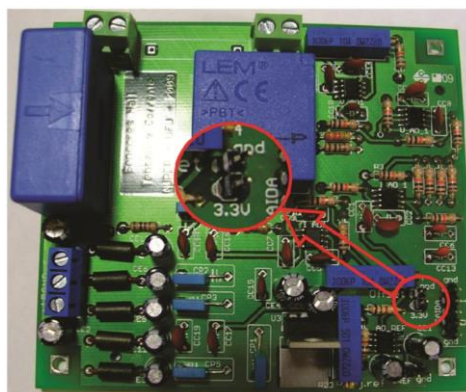


Figura 20: Pinos de medição da tensão 3,3 V.
Fonte: NUPEP (2011).

4. Foi ajustado o valor da tensão lida no pino V.ref em relação ao GND para 1,5 V acionando o potenciômetro mostrado na FIG. 21. Esse valor será somado em ambos os canais para calibração do zero do sinal

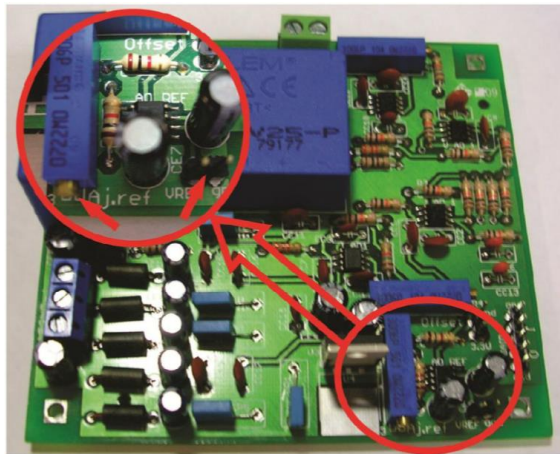


Figura 21: Calibração do zero.
Fonte: NUPEP (2011).

5. Foram calibrados os offsets nos canais do sensor de tensão e de corrente onde, suas respectivas saídas foram ajustadas para o mesmo valor colocado no pino Vref. O ajuste foi feito girando-se o potenciômetro correspondente a cada canal. As FIG. 22 e 23 abaixo mostram qual potenciômetro corresponde a cada canal, bem como os respectivos pinos da saída do canal.

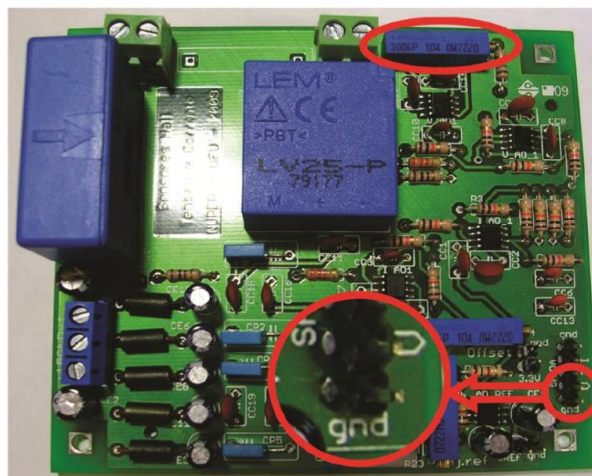


Figura 22: Calibração de offset do canal do sensor de tensão.
Fonte: NUPEP (2011).

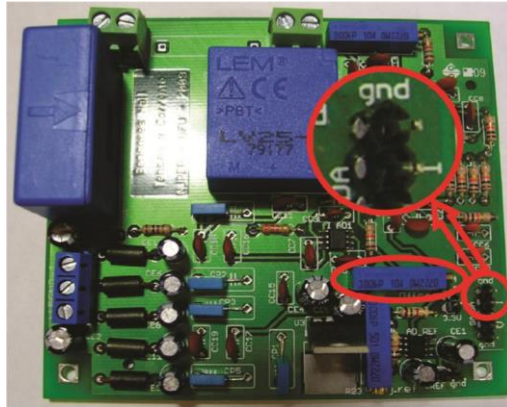


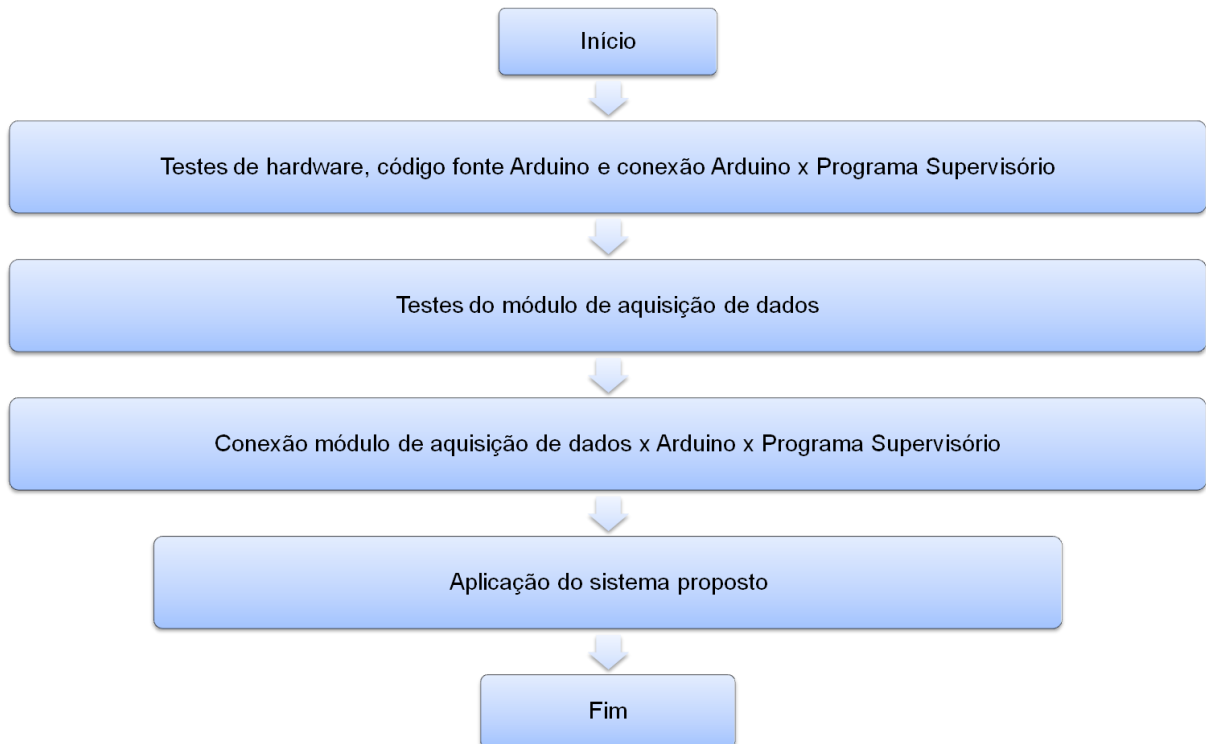
Figura 23: Calibração de offset do canal do sensor de corrente.
Fonte: NUPEP (2011).

O valor do offset ajustado no canal do sensor de corrente foi de 1,514 Volts. Devido a limitações do próprio módulo em questão, não foi possível ajustar para o valor recomendado de 1,5 Volts.

Para aplicação desse módulo de aquisição juntamente com o Arduino, será necessário realizar um ajuste dos valores lidos nas entradas analógicas, devido à elevação do "zero" ao valor de 1,5 Volts, o qual será discutido nos resultados mais adiante.

4 RESULTADOS E DISCUÇÕES

Conforme mostrado na metodologia, a execução desse trabalho foi dividida em etapas distintas, sendo necessária, portanto, a realização de testes nos diferentes momentos, visando garantir o correto funcionamento do projeto proposto. No Fluxograma 2 está representada a evolução do que foi realizado.



Fluxograma 2: Etapas dos resultados da aplicação da metodologia.

A seguir estão descritos os resultados de cada etapa mencionada no Fluxograma 2.

4.1 Testes de hardware, código fonte Arduino e conexão Arduino x Programa Supervisório

Para testar o funcionamento do código fonte criado, das portas de comunicação utilizadas da placa do Arduino Uno e da conexão Arduino x Programa Supervisório, foi conectada, simultaneamente, uma fonte CC em cada entrada analógica do Arduino, fornecendo alimentação limitada em 5 Volts, com o intuito de simular o sinal vindo do módulo de aquisição de dados.

A resolução do conversor AD do Arduino é de 10 bits e, neste caso, uma tensão de 5 Volts, máximo valor analógico permitido, será representada pelo número digital 1023, o que fornece a seguinte resolução:

$$Resolução = \frac{valor_analógico}{valor_digital} = \frac{5}{1023} \cong 4,89 \times 10^{-3} \text{ Volts/bit}$$

No primeiro momento, as tensões aplicadas nas entradas analógicas foram todas de 0 Volts. Esse teste teve o objetivo de verificar se os zeros aplicados nas entradas do Arduino seriam lidos na ferramenta Monitor Serial como zeros também, garantindo, portanto a correta representação do fundo de escala inferior da aplicação. A FIG. 24 mostra, por meio da representação digital dos multímetros, os valores reais aplicados, sendo, da esquerda para a direita, as tensões aplicadas nas portas analógicas definidas no código fonte como `corrente_poli`, `tensao_poli`, `corrente_mono` e `tensao_mono`, respectivamente. Esses valores, apesar de se mostrarem estáticos, oscilavam em escala milivoltmétrica em torno do zero absoluto. Essa oscilação e a não obtenção do zero absoluto podem ser explicadas pela variação espontânea da carga interna da própria fonte e foram consideradas desprezíveis, não impactando no resultado do projeto.



Figura 24: Tensão de 0 Volts aplicada nas entradas analógicas do Arduino.

Os sinais lidos na ferramenta Monitor Serial e mostrados na FIG. 25 estão em escala digital, ou seja, representando valores variando de 0 a 1023 que correspondem à

variação de 0 a 5 Volts aplicados na entrada. Para esse teste, projetaram basicamente os valores que foram aplicados nas entradas do Arduino, onde foi possível observar oscilações que correspondiam às oscilações relatadas anteriormente. Por exemplo, o valor analógico de 0,010 Volts mostrado no ultimo multímetro à direita é equivalente ao valor 2 em representação digital, conforme mostrado a seguir.

$$\text{Valor_digital}_{0,010} = \frac{1023}{5} \times 0,010 = 2$$

tensão_mono	corrente_mono	tensão_poli	corrente_poli
1.00	1.00	1.00	1.00
0.00	1.00	0.00	3.00
1.00	1.00	0.00	1.00
0.00	2.00	1.00	1.00
0.00	0.00	0.00	0.00
0.00	1.00	0.00	0.00
0.00	1.00	0.00	0.00
1.00	1.00	0.00	1.00
0.00	1.00	1.00	1.00
0.00	1.00	1.00	2.00
1.00	1.00	1.00	1.00
1.00	1.00	0.00	0.00
2.00	1.00	1.00	0.00
1.00	1.00	0.00	0.00
0.00	2.00	0.00	0.00
0.00	1.00	0.00	1.00
0.00	2.00	1.00	2.00
0.00	1.00	1.00	2.00
1.00	2.00	0.00	2.00
0.00	1.00	1.00	2.00
0.00	1.00	0.00	2.00
0.00	0.00	0.00	1.00
0.00	0.00	0.00	0.00
1.00	1.00	0.00	0.00
2.00	1.00	0.00	0.00

Figura 25: Leitura do sinal de 0 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando a ferramenta Monitor Serial.

Da mesma forma como foi feito na ferramenta Monitor Serial, os valores também foram lidos por meio do Programa Supervisor, onde também estão representados em escala digital. Conforme pode ser visto na FIG. 26, os valores corresponderam ao que foi proposto.

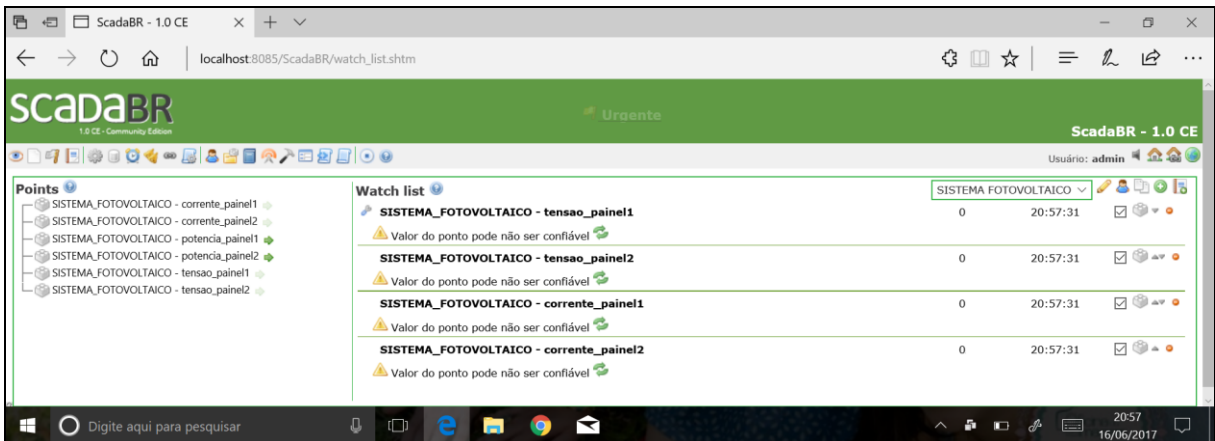


Figura 26: Leitura do sinal de 0 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando o Programa Supervisório.

Para adequação dos valores negativos gerados causados pela oscilação da fonte de tensão, foi necessário inserir no código fonte do Arduino um tratamento, conforme mostrado na FIG. 27, com o objetivo de eliminar a representação desses sinais no Programa Supervisório.

```
// Tratamento para fundo de escala inferior da leitura das portas analógicas não serem menor que zero
if (tensao_mono_analogica < 0)
  tensao_mono_analogica = 0;
if (corrente_mono_analogica < 0)
  corrente_mono_analogica = 0;
if (tensao_poli_analogica < 0)
  tensao_poli_analogica = 0;
if (corrente_poli_analogica < 0)
  corrente_poli_analogica = 0;
```

Figura 27: Parte do código fonte Arduino que trata os valores menores que zero nas entradas analógicas.

Em um segundo momento, foram aplicados 5 Volts simultaneamente em cada entrada analógica do Arduino e as leituras realizadas por meio da ferramenta Monitor Serial e pelo Programa Supervisório também corresponderam ao esperado. As FIG. 28, 29 e 30 mostram o teste realizado. Todos os valores que oscilaram acima de 5 Volts nas entradas analógicas foram representados como 1023.



Figura 28: Tensão de 5 Volts aplicada nas entradas analógicas do Arduino.

tensão_mono	corrente_mono	tensão_poli	corrente_poli
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1022.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00
1023.00	1023.00	1023.00	1023.00

Figura 29: Leitura do sinal de 5 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando a ferramenta Monitor Serial.

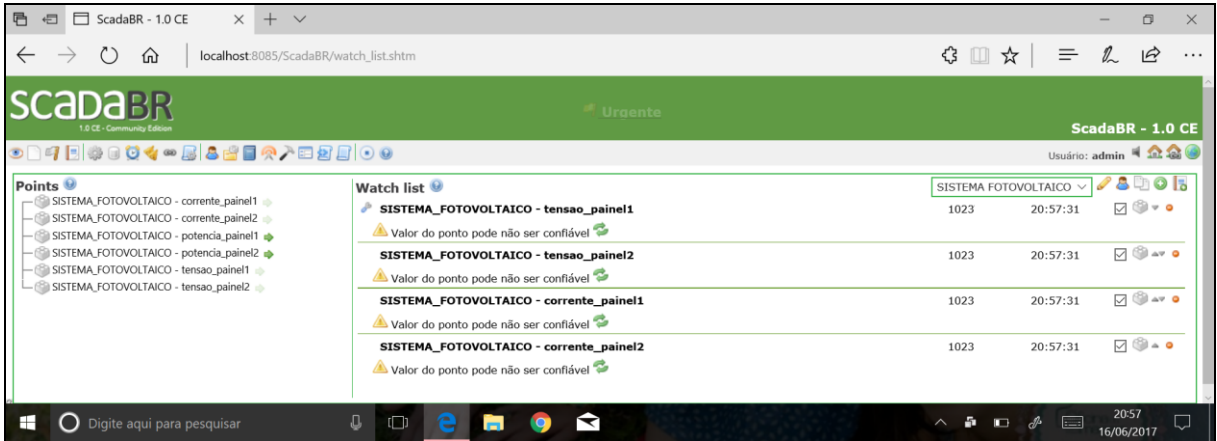


Figura 30: Leitura do sinal de 5 Volts em escala digital, aplicados nas portas de entrada analógica do Arduino, utilizando o Programa Supervisório.

Em um terceiro momento, as tensões aplicadas foram diferentes em cada entrada com o objetivo de validar a representação de valores intermediários. O resultado do teste é apresentado nas FIG. 31, 32 e 33.



Figura 31: Valores de tensão intermediários aplicados nas entradas A3, A2, A1 e A0 do Arduino respectivamente.

tensão_mono	corrente_mono	tensão_poli	corrente_poli
868.00	573.00	333.00	47.00
870.00	573.00	333.00	46.00
870.00	572.00	332.00	45.00
870.00	573.00	332.00	44.00
870.00	573.00	332.00	45.00
870.00	572.00	333.00	46.00
870.00	573.00	333.00	47.00
870.00	573.00	333.00	45.00
870.00	572.00	333.00	46.00
870.00	573.00	332.00	45.00
870.00	572.00	332.00	47.00
870.00	572.00	333.00	47.00
870.00	573.00	332.00	46.00
870.00	573.00	332.00	45.00
870.00	573.00	333.00	46.00
870.00	572.00	332.00	45.00
870.00	572.00	332.00	46.00
870.00	573.00	333.00	46.00
870.00	573.00	331.00	45.00
870.00	572.00	332.00	46.00
870.00	573.00	333.00	46.00
871.00	573.00	332.00	46.00
869.00	573.00	332.00	47.00

Figura 32: Leitura das tensões intermediárias aplicadas nas entradas do Arduino por meio da ferramenta Monitor Serial.

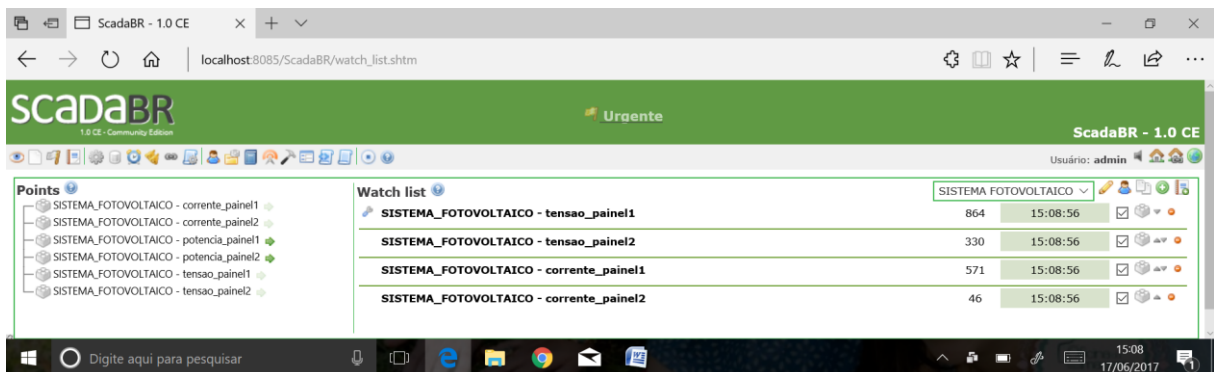


Figura 33: Leitura das tensões intermediárias aplicadas nas entradas do Arduino por meio do Programa Supervisório.

Para validação dos valores intermediários, foi aplicada a equação da resolução da tensão nos resultados obtidos, sendo:

$$\text{Re solução}_{\text{tensão_mono}} = \frac{\text{valor_analógico_tensão_mono}}{\text{valor_digital_tensão_mono}} = \frac{4,259}{870} \cong 4,89 \times 10^{-3} \text{Volts}$$

$$\text{Re solução}_{\text{corrente_mono}} = \frac{\text{valor_analógico_corrente_mono}}{\text{valor_decimal_corrente_mono}} = \frac{2,805}{573} \cong 4,89 \times 10^{-3} \text{Volts}$$

$$\text{Re solução}_{\text{tensão_poli}} = \frac{\text{valor_analógico_tensão_poli}}{\text{valor_decimal_tensão_poli}} = \frac{1,636}{332} \cong 4,93 \times 10^{-3} \text{Volts}$$

$$Resolução_corrente_poli = \frac{valor_analógico_corrente_poli}{valor_decimal_corrente_poli} = \frac{0,234}{46} \cong 5,09 \times 10^{-3} Volts$$

Com os resultados das equações, pôde-se observar que a resolução dos valores lidos nas entradas *tensao_mono* e *corrente_mono* foi idêntica ao valor esperado, porém as resoluções das entradas *tensao_poli* e *corrente_poli* apresentaram erros na conversão que podem ser explicados pela oscilação, já que os mesmos são desprezíveis, conforme cálculos mostrados a seguir:

$$Erro_tensão_poli = \frac{resolução_tensão_poli - resolução_esperada}{resolução_esperada} \times 100 = \frac{4,93 - 4,89}{4,89} \times 100 \cong 0,82\%$$

$$Erro_corrente_poli = \frac{resolução_corrente_poli - resolução_esperada}{resolução_esperada} \times 100 = \frac{5,09 - 4,89}{4,89} \times 100 \cong 4,09\%$$

Em um último momento deste teste, foi simulada uma variação da tensão aplicada nos quatro canais, variando de 1 em 1, a partir de 0 até 5 Volts e, do resultado, foi gerado um relatório, mostrado anteriormente na FIG. 17, confirmando portanto, a efetividade dos resultados e a validação do correto funcionamento do código fonte, das portas de comunicação utilizadas e da conexão Arduino x Programa Supervisor.

4.2 Testes do módulo de aquisição de dados

Para realização desse teste, primeiramente foi montada uma associação de fontes de tensão, limitada em 350 Volts, com o objetivo de se obter um sinal simulando o que seria gerado nas placas solares do sistema de cogeração a que esse trabalho foi proposto. O sinal de tensão desse conjunto de fontes foi aplicado nas duas entradas do módulo de aquisição de dados, sendo que o teste foi feito em um canal de cada vez.

Aplicando o sinal da associação de fontes na entrada do canal de tensão e variando-o de 0 a 350 Volts, foi obtido uma resposta na saída do respectivo canal que variou de 1,5 a 2,02 Volts. Nesse caso, o valor de 1,5 Volts representa o "zero", sendo necessário, portanto, subtraí-lo do valor de 2,02 Volts para identificar a verdadeira representação do valor máximo de 350 Volts.

Para sensibilizar o sensor hall do canal de corrente, foi necessário enlaçar o cabo positivo do conjunto de fontes no mesmo. O Quadro 2 apresenta a relação quantidade de voltas do terminal da fonte no sensor por resposta na saída do canal para uma corrente de 1 A aplicada na entrada. Perceba que para cada volta do terminal enlaçada no dispositivo, a resposta na saída é somada de 30 mVolts.

Para realização do teste foram dadas 6 voltas do terminal da fonte no sensor.

QUANTIDADE DE VOLTAS DO TERMINAL DA FONTE NO SENSOR DE CORRENTE	RESPOSTA NA SAÍDA DO CANAL DE CORRENTE (EM Volts)
1	1,574
2	1,604
3	1,634
4	1,664
5	1,694
6	1,724

Quadro 2: Voltas do terminal da fonte no sensor x sinal na saída do canal de corrente.

Foi necessário também, para esse teste, instalar uma carga na linha desse mesmo condutor, visando possibilitar o fluxo de corrente pelo sensor. Para isso, um resistor de potência com 350 ohms de resistência foi utilizado, o qual proporcionou um fluxo de corrente limitado em 1 A, conforme equação a seguir.

$$Corrente_sensor = \frac{Tensão_total_aplicada}{Carga_na_linha_do_sensor} = \frac{350_Volts}{350_ohms} = 1A$$

Para uma tensão variando de 0 a 350 Volts aplicados na entrada, a corrente que passou pelo sensor variou de 0 a 1 A e conseqüentemente, o sinal da saída do canal de corrente variou de 1,514 a 1,724 Volts. Da mesma forma como no canal de tensão, o valor de 1,514 Volts representa o zero e deverá ser subtraído do valor de 1,724 Volts para identificação do valor que representa 1 A que foi a corrente máxima que passou pelo sensor.

Os testes foram realizados em apenas um módulo de aquisição de dados devido à indisponibilidade de tal dispositivo no CEFET - MG / Unidade Araxá em perfeitas condições de funcionamento.

4.3 Conexão módulo aquisição de dados x Arduino x Programa Supervisório

Como mencionado anteriormente, após a conversão A/D dos valores lidos nas entradas analógicas, eles são disponibilizados na saída do Arduino na forma digital, sendo necessário, portanto, realizar uma nova conversão de forma a restabelecer o valor que verdadeiramente representa o valor que foi aplicado nas entradas.

Essa conversão foi feita inserindo, no código fonte Arduino, os fatores operacionais destinados a cada canal, conforme descrito nas FIG. 34 e 35.

```
// Definição dos valores do zero e fatores de conversão
zero_tensao = 1023 / 5 * 1.5;           // Valor em binário
zero_corrente = 1023 / 5 * 1.514;      // Valor em binário
fator_tensao = 350 / (1023 / 5 * (2.02 - 1.5)); // Valor em decimal
fator_corrente = 9 / (1023 / 5 * (3.404 - 1.514)); // Valor em decimal
```

Figura 34: Parte do código fonte que define os fatores operacionais a serem aplicados no sinal de entrada dos canais do módulo de aquisição de dados.

Observando a FIG. 34, os resultados das equações que estão sendo armazenados nas variáveis `zero_tensao` e `zero_corrente` são valores digitais equivalentes ao valor analógico setado nas saídas dos canais de tensão e corrente, respectivamente. A representação desses valores em forma digital é necessária para que a subtração dos mesmos no momento da conversão seja feita na escala correta.

Já os resultados das equações que estão sendo armazenados nas variáveis `fator_tensao` e `fator_corrente` são valores digitais dos fatores de conversão dos canais de tensão e corrente respectivamente e têm a função de garantir que o range de valores aplicados nas entradas do módulo de aquisição de dados sejam representados de forma idêntica no Programa Supervisório proposto.

No caso do canal de tensão, a variação de 0 a 350 Volts aplicados na entrada do módulo se equivale à variação de 1,5 a 2,02 Volts na saída do mesmo e conseqüentemente a uma variação de 307 a 413 em escala digital na saída do

conversor A/D do Arduino. Os valores em escala digital disponibilizados depois da conversão A/D são obtidos conforme equações a seguir:

$$\text{Zero}_{\text{digital}} = \frac{1023}{5} \times 1,5 = 307$$

$$\text{Tensão}_{\text{máxima}_{\text{digital}}} = \frac{1023}{5} \times 2,02 = 413$$

Sendo assim, uma variação de 350 Volts na entrada do canal de tensão equivale a uma variação de 0,52 Volts em sua saída que equivale um range digital na saída do Arduino de 106, conforme mostrado nas equações a seguir:

$$\text{Range}_{\text{na}_{\text{entrada}_{\text{do}_{\text{canal}_{\text{de}_{\text{tensão}}}}} = 350 - 0 = 350\text{Volts}}$$

$$\text{Range}_{\text{na}_{\text{saída}_{\text{do}_{\text{canal}_{\text{de}_{\text{tensão}}}}} = 2,02 - 1,5 = 0,52\text{Volts}}$$

$$\text{Range}_{\text{digital}_{\text{na}_{\text{saída}_{\text{do}_{\text{Arduino}}}}} = 413 - 307 = 106$$

Para o canal de corrente, a variação de 0 a 1 A aplicados na entrada do módulo se equivale à variação de 1,514 a 1,724 Volts na saída do mesmo e conseqüentemente a uma variação de 310 a 353 em escala digital na saída do conversor A/D do Arduino. Os valores em escala digital disponibilizados depois da conversão A/D são obtidos conforme equações a seguir:

$$\text{Zero}_{\text{digital}} = \frac{1023}{5} \times 1,514 = 310$$

$$\text{Corrente}_{\text{máxima}_{\text{digital}}} = \frac{1023}{5} \times 1,724 = 353$$

Da mesma forma, uma variação de 1 A na entrada do canal de corrente equivale a uma variação de 0,21 Volts em sua saída que equivale um range digital na saída do Arduino de 43, conforme mostrado nas equações a seguir:

$$\text{Range}_{\text{na}_{\text{entrada}_{\text{do}_{\text{canal}_{\text{de}_{\text{corrente}}}}} = 1 - 0 = 1\text{A}}$$

$$\text{Range}_{\text{na}_{\text{saída}_{\text{do}_{\text{canal}_{\text{de}_{\text{corrente}}}}} = 1,724 - 1,514 = 0,21\text{Volts}}$$

$$\text{Range}_{\text{digital}_{\text{na}_{\text{saída}_{\text{do}_{\text{Arduino}}}}} = 353 - 310 = 43$$

Fazendo a projeção dos cálculos para uma corrente de 9 A, que será o máximo valor gerado no sistema fotovoltaico em questão, tem-se os seguintes ranges:

$$\text{Range}_{na_entrada_do_canal_de_corrente} = 9A$$

$$\text{Range}_{na_saída_do_canal_de_corrente} = 9 \times 0,21 = 1,89\text{Volts}$$

$$\text{Range}_{digital_na_saída_do_Arduino} = 9 \times 43 = 387$$

Como pode ser visualizado na FIG. 35, os valores lidos nas entradas analógicas do Arduino são subtraídos dos valores dos respectivos zeros, multiplicados pelos fatores de conversão correspondentes a cada tipo de canal e, em seguida armazenados em variáveis para posteriormente serem enviados ao Programa Supervisório.

Especificamente nos canais de corrente, foi necessário multiplicar o resultado da operação por 1000, visto que o Programa Supervisório somente faz leitura de números inteiros e, neste caso, valores menores que 1 A não seriam representados.

```
// Ajuste dos dados lidos nas entradas analógicas e direcionamento para as variáveis declaradas
tensao_mono_analogica = (analogRead(tensao_mono) - zero_tensao) * fator_tensao;
corrente_mono_analogica = (analogRead(corrente_mono) - zero_corrente) * fator_corrente * 1000;
tensao_poli_analogica = (analogRead(tensao_poli) - zero_tensao) * fator_tensao;
corrente_poli_analogica = (analogRead(corrente_poli) - zero_corrente) * fator_corrente * 1000;
```

Figura 35: Parte do código fonte que aplica os fatores operacionais nos sinal de entrada dos canais do módulo de aquisição de dados.

As FIG. 36, 37, 38 e 39 mostram por meio da ferramenta Monitor Serial e do Programa Supervisório, o resultado da aplicação dos fatores operacionais no sinal de 350 Volts inserido no canal de tensão, durante os testes de conexão do módulo de aquisição de dados x Arduino x Programa Supervisório.

tensão_mono	corrente_mono	tensão_poli	corrente_poli
102.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
101.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
102.00	0.00	0.00	0.00
103.00	0.00	0.00	0.00

Figura 36: Leitura do sinal digital do canal de tensão abatendo o zero, utilizando a ferramenta Monitor Serial.

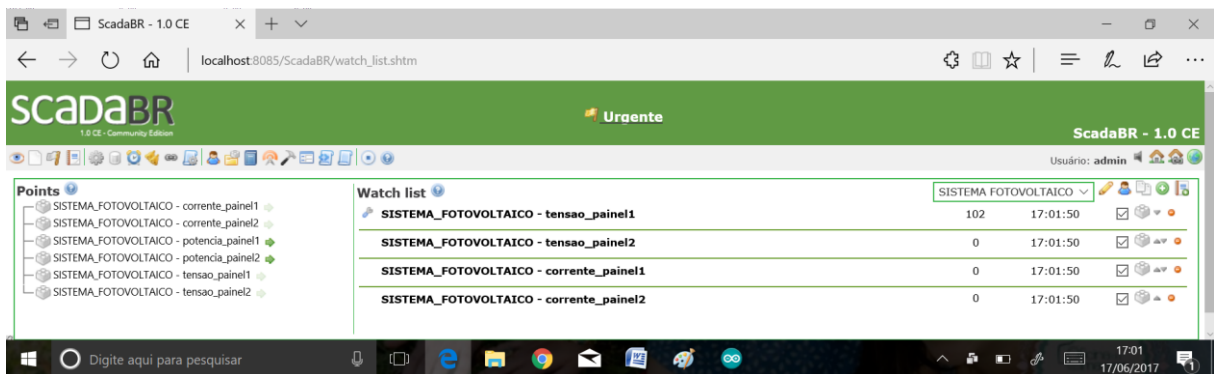


Figura 37: Leitura do sinal digital do canal de tensão abatendo o zero, utilizando o Programa Supervisório.

Com o resultado, pode-se perceber que o valor em torno de 102 mostrado nas FIG. 36 e 37 tem um erro já que o valor esperado seria de 106. Esse erro é calculado a seguir e, quando convertido para representação em tensão, mostra um resultado expressivo que precisa ser ajustado.

$$\text{Erro}_{\text{canal}_{\text{tensão}}} = \frac{\text{valor}_{\text{esperado}}}{\text{resultado}} = \left(1 - \frac{106}{102}\right) \times 100 = 3,92\%$$

$$\text{Re}_{\text{presentação}_{\text{do}_{\text{erro}}}} = \frac{\text{tensão}_{\text{máxima}} \times \text{erro}}{100} = \frac{350 \times 3,92}{100} = 13,73 \text{Volts}$$

A existência do erro foi também comprovada nos resultados seguintes, onde finalmente foram aplicados os fatores de conversão. Conforme mostrado nas FIG. 38 e 39, o erro não foi exatamente de 13,73 Volts, mas valores em torno deste.

tensão_mono	corrente_mono	tensão_poli	corrente_poli
333.24	0.00	0.00	0.00
336.54	0.00	0.00	0.00
333.24	0.00	0.00	0.00
336.54	0.00	0.00	0.00
346.44	0.00	0.00	0.00
339.84	0.00	0.00	0.00
343.14	0.00	0.00	0.00
336.54	0.00	0.00	0.00
333.24	0.00	0.00	0.00
339.84	0.00	0.00	0.00
336.54	0.00	0.00	0.00
336.54	0.00	0.00	0.00
333.24	0.00	0.00	0.00
336.54	0.00	0.00	0.00
333.24	0.00	0.00	0.00
343.14	0.00	0.00	0.00
339.84	0.00	0.00	0.00
346.44	0.00	0.00	0.00
326.64	0.00	0.00	0.00
329.94	0.00	0.00	0.00
333.24	0.00	0.00	0.00
333.24	0.00	0.00	0.00
329.94	0.00	0.00	0.00
336.54	0.00	0.00	0.00
343.14	0.00	0.00	0.00

Figura 38: Leitura do sinal do canal de tensão com aplicação dos fatores de conversão, utilizando a ferramenta Monitor Serial.

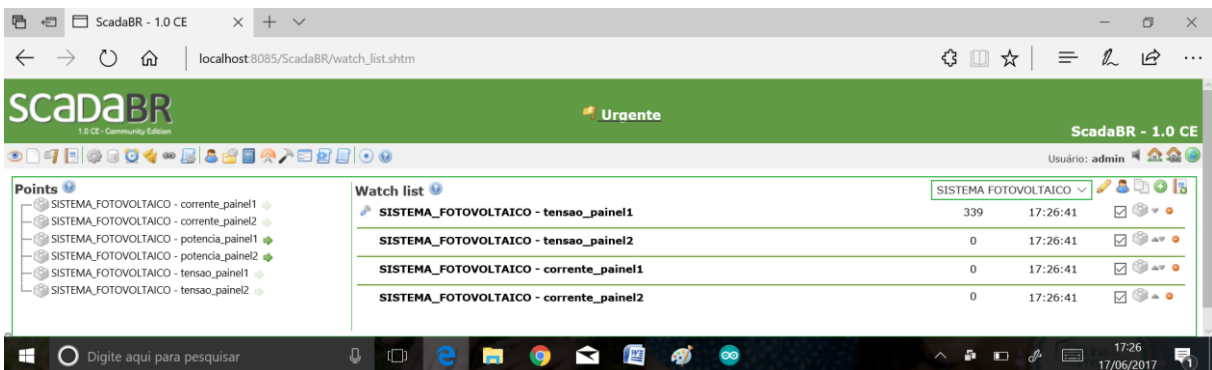


Figura 39: Leitura do sinal do canal de tensão com aplicação dos fatores de conversão, utilizando o Programa Supervisório.

Da mesma forma como feito no canal de tensão, foi obtido o resultado aplicando somente a subtração do zero no sinal de 1 A inserido no canal de corrente, conforme estão demonstrados nas FIG. 40, 41, 42 e 43.

tensãõ_mono	corrente_mono	tensãõ_poli	corrente_poli
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.99	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.99	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.95	0.00	0.00
0.00	0.97	0.00	0.00
0.00	0.97	0.00	0.00

Figura 42: Leitura do sinal do canal de corrente com aplicação dos fatores de conversão, utilizando a ferramenta Monitor Serial.

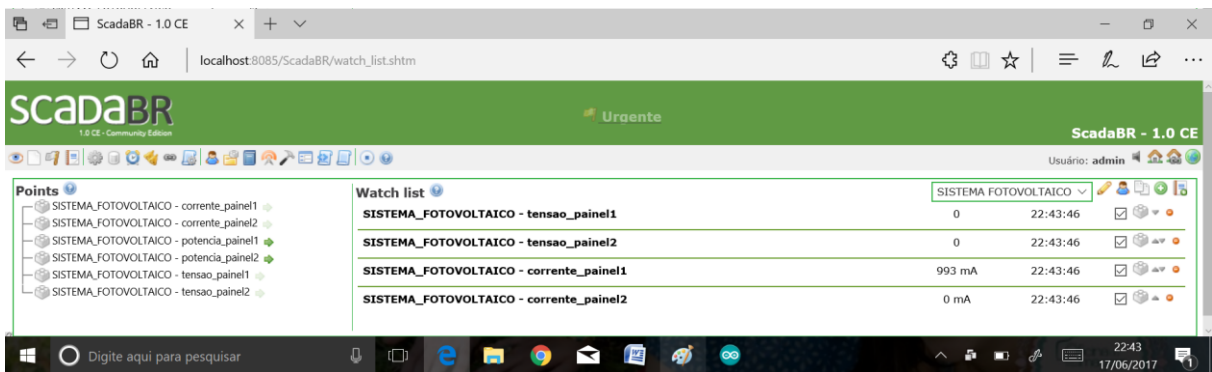


Figura 43: Leitura do sinal do canal de corrente com aplicação dos fatores de conversão, utilizando o Programa Supervisório.

O erro na leitura do canal de corrente, após a aplicação do fator de conversão, é calculado a seguir e pode ser desconsiderado praticamente desprezível, já que não tem significância expressiva no resultado.

$$Erro_canal_corrente = \frac{valor_esperado}{resultado} = \left(1 - \frac{1}{0,993}\right) \times 100 = 0,7\%$$

$$Re\ apresentação_do_erro = \frac{corrente_máxima \times erro}{100} = \frac{1 \times 0,7}{100} = 7mA$$

4.4 Aplicação do sistema proposto

Para finalizar o desenvolvimento desta pesquisa, foi conectado ao sistema desenvolvido uma micro unidade de geração de energia fotovoltaica composta de dois módulos de Silício policristalino conectados em série, previamente instalada nas mediações do laboratório de eletrônica do CEFET - MG / Unidade Araxá, para monitoramento dos sinais de tensão, corrente e potencia gerados por ela.

As especificações técnicas dos módulos que compõe a unidade de geração são apresentadas na FIG. 44.




 WARNING			
 HAZARDOUS ELECTRICITY CAN SHOCK, BURN OR CAUSE DEATH. DO NOT TOUCH TERMINALS.			
PHOTOVOLTAIC MODULE			
MODEL	KC40T		
SER NO.	06Z58H1489		
DATE	2006.12		
			T
IRRADIANCE AND CELL TEMPERATURE	1000Wm ⁻² AM 1.5 25 °C	800Wm ⁻² AM 1.5 47 °C	MAX. SYS VOLT. 600 V
P _{max}	43W	30W	SERIES FUSE 6 A
V _{pmax}	17.4 V	15.3 V	MASS 4.5 kg
I _{pmax}	2.48 A	1.97 A	
V _{oc}	21.7 V	—	
I _{sc}	2.65 A	—	
 UL US LISTED 9P82	FIELD WIRING		FIRE RATING
	STRANDED COPPER ONLY 10 ~14 AWG INSULATED FOR 90°C		CLASS C

Figura 44: Especificações técnicas dos módulos fotovoltaicos instalados na micro unidade de geração de energia fotovoltaica nas mediações do laboratório de eletrônica do CEFET-MG / Unidade Araxá.

A carga conectada na unidade de geração de energia solar foi de 35 Ohms, a qual possibilita uma corrente limitada em no máximo 0,99 A, conforme equação abaixo:

$$Corrente = \frac{V_{p\max_m\acute{o}dulo_1} + V_{p\max_m\acute{o}dulo_2}}{35} = \frac{17,4 + 17,4}{35} = 0,99A$$

O monitoramento foi realizado no dia 25 de junho no período de 15h30min as 17h20min e do resultado foi gerado um relatório que é mostrado na FIG. 45.

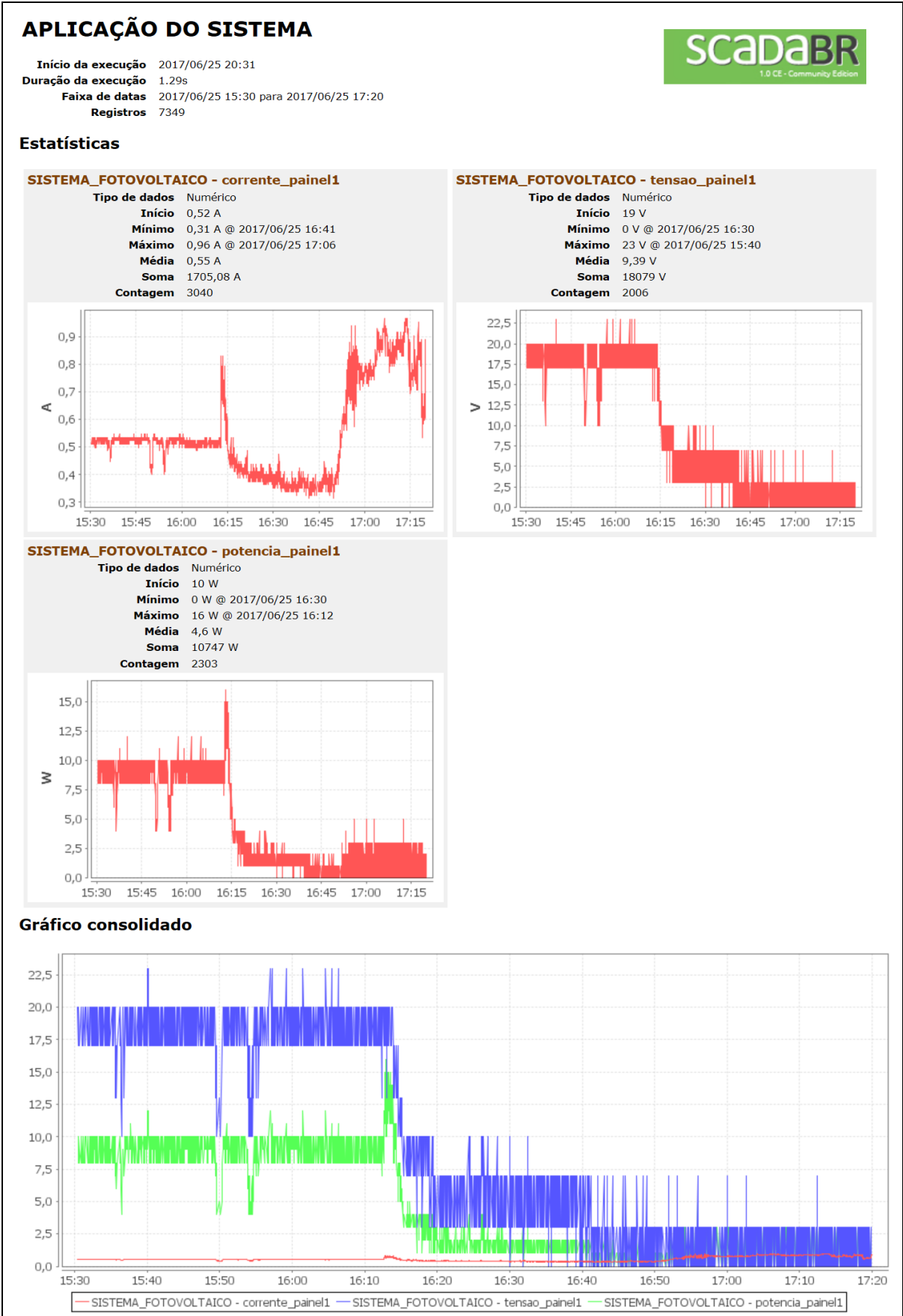


Figura 45: Relatório de monitoramento de geração de energia fotovoltaica.

CONCLUSÕES

O Programa Supervisório desenvolvido neste trabalho de pesquisa mostrou-se uma ferramenta com interface amigável e de fácil utilização, com opções de monitoramento por meio de duas diferentes telas de supervisão, além de possibilitar a visualização de dados históricos e geração de relatórios.

O sistema de aquisição de dados, por sua vez, também cumpriu seu papel. Apesar da disponibilidade de apenas um módulo de aquisição de dados para realização deste trabalho, todos os quatro canais foram testados e o correto funcionamento de ambos foi validado conforme resultados mostrados anteriormente.

Os erros de leitura ocorridos durante os testes realizados não comprometem o resultado final do trabalho, visto que eles são desprezíveis e podem ser ajustados posteriormente. Uma atenção maior deve ser dada para o canal do sensor de tensão do módulo de aquisição, o qual, além de apresentar erro de leitura nos testes, sua resposta na aplicação do sistema mostrou uma oscilação no sinal que não retrata o real comportamento da tensão gerada, conforme pode ser visto no gráfico SISTEMA_FOTOVOLTAICO - tensao_painel1 da FIG. 45.

De uma forma geral, os objetivos foram alcançados, mediante o desenvolvimento um sistema de aquisição de dados que se utiliza de uma plataforma micro processada de livre acesso e um módulo de sensores de efeito Hall previamente disponibilizado pelo CEFET - MG / Unidade Araxá, assim como o desenvolvimento de um programa supervisório, utilizando software livre, para monitoramento da energia que será gerada pela unidade fotovoltaica a ser implantada nas mediações da referida instituição, atendendo, de forma integral e satisfatória, ao que foi proposto inicialmente.

5 TRABALHOS FUTUROS

Como possíveis trabalhos futuros, pode-se apontar:

- Desenvolvimento de um módulo de aquisição de dados, em substituição ao que foi utilizado para realização deste trabalho, que atenda especificamente a aplicação de coletar sinais de tensão e corrente na saída de sistemas de geração de energia fotovoltaica;
- Desenvolver a transmissão do sinal coletado nos sistemas de geração de energia fotovoltaica sem a utilização de meio físico (transmissão WIFI);
- Desenvolver uma ferramenta para disponibilizar as informações da geração de energia fotovoltaica na WEB.

REFERÊNCIAS

- ANEEL - AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. Chamada nº 013/2011: **Projeto estratégico: arranjos técnicos e comerciais para inserção da geração solar fotovoltaica na matriz energética brasileira.** Brasília, 2011.
- ANEEL - AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. Ren 481: **Alteração dos procedimentos de redução de tarifas de uso dos sistemas elétricos de transmissão e distribuição para empreendimentos de geração de energia renovável da Ren 77.** Brasília, 2012a.
- ANEEL - AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. Ren 482: **Condições gerais para acesso de microgeração e minigeração distribuída aos sistemas de distribuição de energia elétrica.** Brasília, 2012b.
- ANEEL - AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. Ren 687/15: **Alteração condições gerais para acesso de microgeração e minigeração distribuída aos sistemas de distribuição de energia elétrica da Ren 482.** Brasília, 2015.
- ARDUINO. **O que é Arduino.** [s.l.: s.n.] 2017a. Disponível em: <<https://www.arduino.cc/en/guide/introduction>>. Acesso em: mar. 2017.
- ARDUINO. **Software Arduino (ide).** [s.l.: s.n.] 2017b. Disponível em: <<https://www.arduino.cc/en/guide/environment>>. Acesso em: mar. 2017.
- ARDUINO. **Visão geral Arduino uno.** [s.l.: s.n.] 2017c. Disponível em: <<https://www.arduino.cc/en/main/arduinoboarduno>>. Acesso em: mar. 2017.
- CENTRO DE REFERÊNCIAS PARA ENERGIA SOLAR E EÓLICA SÉRGIO DE SALVO BRITO - CRESEB. **Tutorial de energia solar fotovoltaica.** Rio de Janeiro: s.n., 2008.
- WIKIPÉDIA. **Efeito hall.** [s.l. S.n.] Jun. 2017. Disponível em: <https://pt.wikipedia.org/wiki/efeito_hall>. Acesso em: 05 jul. 2017.
- EMBEDDED ARCHITECTS TECNOLOGIA. **O que é um sistema embarcado** [s.l., s d.]. Disponível em: <<http://www.embarc.com.br/p1600.aspx>>. Acesso em: 22 set. 2017.
- FERREIRA, R. R.; SILVA FILHO, P. C. Da. **Energia solar fv – geração de energia limpa.** Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte - Universidade Federal do Rio Grande do Norte, Departamento de Física Teórica e Experimental Rio Grande do Norte, 2010. Disponível em: <<http://connepi.ifal.edu.br/ocs/index.php/connepi/connepi2010/paper/viewfile/1828/1058>> acesso em: set. 2016.
- FONTOURA, K. L. **Projeto e instalação de uma unidade de microgeração distribuída baseada na energia fotovoltaica com vistas à formação de recursos humanos e ao incentivo à utilização de sistemas fotovoltaicos conectados à rede elétrica como ação de eficiência energética.** Projeto de pesquisa CNPQ, CEFET - Campus Araxá. Araxá, 2015.

MODBUS ORGANIZATION INC. **Modbus application protocol specification v1.1b3**. [s.l.], 26 abr. 2012. Disponível em: <http://www.modbus.org/docs/modbus_application_protocol_v1_1b3.pdf>. Acesso em: jun. 2017.

NIEDZIALKOSKI, R. K. **Desempenho de painéis solares mono e policristalinos em um sistema de bombeamento de água**. Dissertação (Mestrado) – Universidade Estadual do Oeste do Paraná, Cascavel, 2013. Disponível em: <http://www4.unioeste.br/portalpos/media/dissertacao_rosana_k_niedzialkoski.pdf>. Acesso em: set. 2016.

NÚCLEO DE PESQUISA EM ELETRÔNICA DE POTÊNCIA - NUPEP. **Módulos de aquisição e condicionamento de sinais de tensão e corrente**. Universidade Federal de Uberlândia, Uberlândia/MG, 2011. 5 p.

OKADA, K. F. A. **Software para monitoramento da curva de potência em painel solar utilizando microcontrolador Pic na cidade de Araxá**. Projeto (Iniciação Científica Jr., CNPQ; FAPEMIG) - CEFET - Campus Araxá, Araxá, 2013.

PINHO, J. T.; GALDINO, M. A. (Org.). **Manual de engenharia para sistemas fotovoltaicos**. Rio de Janeiro: CEPEL - CRESESB, 2014. Disponível em: <http://www.cresesb.cepel.br/publicacoes/download/manual_de_engenharia_fv_2014.pdf>. Acesso em: set. 2016.

PORTAL - Energias Renováveis. **Principais tipos de células fotovoltaicas constituintes de painéis solares**. [s.l.: s.n.] 2011. Disponível em: <<http://www.portal-energia.com/principais-tipos-de-celulas-fotovoltaicas-constituintes-de-paineis-solares/>>. Acesso em: set. 2016.

SCADABR. Automação Open Source. **Sobre o SCADABR**. [s.l.], 2012. Disponível em: <<http://www.scadabr.org.br/?q=node/1>>. Acesso em: out. 2016.

SCADABR. ScadaBR para Supervisão e Controle. Manual do Software. out. 2010. Disponível em: <<http://www.scadabr.com.br/index.php/2017/06/06/2939/>>. Acesso em: jun.2017.

SOUSA, R. Di. **Os sistemas de energia solar fotovoltaica** - livro digital de introdução aos sistemas solares. Ribeirão Preto: Bluesol Energia Solar, [2016?]. Disponível em: <<http://programaintegradoronline.com.br/wp-content/uploads/2016/03/livro-digital-de-introducao-aos-sistemas-solares-novo.pdf>>. Acesso em: set. 2016.

ANEXO A

ARDUINO UNO

Arduino UNO, representado pela figura a seguir, é uma placa de microcontrolador baseada no processador ATmega328P que possui 14 pinos de I/O (entrada/saída) digitais dos quais 6 podem ser usados como saídas Modulação por Largura de Pulso (PWM), 6 entradas analógicas, um cristal de quartzo de 16 MHz, uma conexão USB, uma tomada de força e um botão de reinicialização. Ele contém o necessário para suportar o microcontrolador bastando apenas conectá-lo a um computador com um cabo USB ou ligá-lo com um adaptador AC-DC ou com bateria (ARDUINO, 2017).



Figura: Placa Arduino UNO.
Fonte: Adaptado de ARDUINO (2017c).

A Tabela a seguir relaciona as especificações técnicas do Arduino UNO.

Microcontrolador	ATmega328P
Tensão operacional	5V
Tensão de entrada (recomendado)	7-12V
Tensão de entrada (limite)	6-20V
Digital I / O Pins	14 (dos quais 6 fornecem saída PWM)
PWM Digital I / O Pins	6
Analog Input Pins	6
Corrente DC por pino de E / S	20 mA
Corrente de CC para o pino de 3.3V	50 mA
Memória flash	32 KB (ATmega328P) dos quais 0,5 KB usado pelo bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Velocidade do relógio	16 MHz
LED_BUILTIN	13
comprimento	68,6 mm
Largura	53,4 mm
Peso	25 g

Tabela 1: Especificações técnicas do Arduino UNO.
Fonte: ARDUINO (2017).

FONTE DE ALIMENTAÇÃO

A placa Arduino Uno pode ser alimentada através da conexão USB ou com uma fonte de alimentação externa. A fonte de alimentação é selecionada automaticamente (ARDUINO, 2017).

A fonte de alimentação externa pode vir de um AC/DC adaptador ou bateria. O adaptador pode ser conectado conectando um plug positivo de centro de 2,1 mm na tomada de alimentação da placa. As derivações de uma bateria podem ser inseridas nos conectores GND e Vin do conector POWER (ARDUINO, 2017).

A placa pode operar em uma fonte externa de 6 a 20 V (Volts). Se fornecido com menos de 7 V, no entanto, o pino 5 V pode fornecer menos de cinco volts e a placa pode ficar instável. Se usar mais de 12 V, o regulador de tensão pode superaquecer e danificar a placa. A faixa recomendada é de 7 a 12 V (ARDUINO, 2017).

Os pinos de alimentação são os seguintes:

- *Vin*: A tensão de entrada para a placa Arduino quando está usando uma fonte de alimentação externa (em oposição a 5 volts da conexão USB ou outra fonte de alimentação regulada). Você pode fornecer a tensão através deste pino, ou, se fornecer a tensão através da tomada de força, acessá-la através deste pino.
- *5V*: Este pino produz um 5 V regulado do regulador na placa. A placa pode ser fornecida com alimentação a partir da tomada de alimentação CC (7 - 12 V), o conector USB (5 V) ou o pino VIN da placa (7-12 V). A tensão de alimentação através dos pinos de 5 V ou 3,3 V ignora o regulador e pode danificar a placa.
- *3V3*: Uma fonte de 3.3 V gerada pelo regulador de bordo. O consumo máximo de corrente é de 50 mA (mili Amperes).
- *GND*: Pinos de aterramento.
- *IOREF*: Este pino na placa Arduino fornece a referência de tensão com a qual o microcontrolador opera. Um escudo configurado corretamente pode ler a tensão do pino IOREF e selecionar a fonte de alimentação apropriada ou habilitar os tradutores de tensão nas saídas para trabalhar com o 5 V ou 3.3 V.

MEMÓRIA

O ATmega328 tem 32 KB (Kilo Bytes) com 0,5 KB ocupado pelo bootloader. Ele também tem 2 KB de Memória de Acesso Estático Aleatório (SRAM) e 1 KB de Memória de Somente Leitura Programável Apagável Eletricamente (EEPROM) que pode ser lido e escrito com a biblioteca EEPROM (ARDUINO, 2017).

ENTRADA E SAÍDA

Cada um dos 14 pinos digitais no Uno pode ser usado como entrada ou saída, usando as funções `pinMode ()`, `digitalWrite ()` e `digitalRead ()`. Eles operam a 5 V. Cada pino pode fornecer ou receber 20 mA como condição de operação recomendada e tem um resistor pull-up interno (desconectado por padrão) de 20-50 KΩ. Um máximo de 40 mA é o valor que não deve ser excedido em qualquer pino de E/S para evitar danos permanentes ao microcontrolador (ARDUINO, 2017).

Além disso, alguns pinos têm funções especializadas:

- *Serial: 0 (RX) e 1 (TX)*: Usado para receber (RX) e transmitir (TX) dados em série.
- *Interrupções Externas: 2 e 3*. Estes pinos podem ser configurados para disparar uma interrupção em um valor baixo, uma borda ascendente ou descendente ou uma alteração no valor.
- *PWM 3, 5, 6, 9, 10 e 11*: Fornece saída PWM de 8 bits com a função `analogWrite ()`.
- *SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK)*: Esses pinos suportam comunicação SPI usando a biblioteca SPI.
- *LED: 13*: Há um built-in LED acionado por pino digital 13. Quando o pino é valor HIGH, o LED está ligado, quando o pino é *low*, ele está desligado.
- *TWI: pino A4 ou SDA e pino A5 ou SCL*: Suporte a comunicação TWI usando a biblioteca Wire.

O Arduino Uno tem 6 entradas analógicas, rotuladas de A0 a A5, cada uma fornecendo 10 bits de resolução (ou seja, 1024 valores diferentes). Por padrão, eles medem de terra para 5 V, embora seja possível alterar a extremidade superior de seu intervalo usando o pino AREF e a função `analogReference ()` (ARDUINO, 2017).

Há um par de outros pinos na placa:

- *AREF*: Tensão de referência para as entradas analógicas. Usado com `analogReference ()`.
- *Restabelecer*: Coloque esta linha low para reiniciar o microcontrolador. Usado tipicamente para adicionar um botão da restauração aos protetores que obstruam um na placa.

COMUNICAÇÃO

O Arduino UNO possui várias facilidades para se comunicar com um computador, outra placa Arduino ou outros microcontroladores. O ATmega328 fornece a comunicação serial Receptor/Transmissor Universal Assíncrono (UART) (5V), que está disponível nos pinos digitais 0 (RX) e 1 (TX). Um ATmega16U2 na placa canaliza esta comunicação serial via USB e aparece como uma porta virtual para software no computador. O firmware 16U2 usa os drivers USB padrão e nenhum driver externo é necessário. O software Arduino inclui um monitor serial que permite que dados simples de texto sejam enviados de e para a placa. Os LEDs RX e TX na placa piscarão quando os dados estiverem sendo transmitidos através do chip USB para serial e conexão USB para computador (mas não para comunicação serial nos pinos 0 e 1) (ARDUINO, 2017).

Uma biblioteca `SoftwareSerial` permite comunicação serial em qualquer um dos pinos digitais do Uno (ARDUINO, 2017).