



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**IVONNE MONTERO DUPONT**

**SISTEMA EMBARCADO LINUX PARA MONITORAMENTO EM  
NUVEM APLICADO A BOMBEAMENTO FOTOVOLTAICO**

**FORTALEZA**

**2017**

IVONNE MONTERO DUPONT

SISTEMA EMBARCADO LINUX PARA MONITORAMENTO EM  
NUVEM APLICADO A BOMBEAMENTO FOTOVOLTAICO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Energias Renováveis.

Orientador: Prof. Dr. Paulo Cesar Marques de Carvalho.

Coorientador: Prof. Dr. Sandro César Silveira Jucá.

FORTALEZA

2017

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

D945s Dupont, Ivonne Montero.  
Sistema embarcado Linux para monitoramento em nuvem aplicado a bombeamento fotovoltaico /  
Ivonne Montero Dupont. – 2017.  
148 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-  
Graduação em Engenharia Elétrica, Fortaleza, 2017.

Orientação: Prof. Dr. Paulo Cesar Marques de Carvalho.  
Coorientação: Prof. Dr. Sandro César Silveira Jucá.

1. Aquisição de Dados. 2. Energias Renováveis. 3. Monitoramento Online. I. Título.

CDD 621.3

---

**IVONNE MONTERO DUPONT**

**SISTEMA EMBARCADO LINUX PARA MONITORAMENTO EM  
NUVEM APLICADO A BOMBEAMENTO FOTOVOLTAICO**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Energias Renováveis.

Aprovada em: 20/02/2017

**BANCA EXAMINADORA**

---

Prof. Dr. Paulo Cesar Marques de Carvalho (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Sandro César Silveira Jucá (Co-orientador)  
Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)

---

Prof. Dr. Paulo Peixoto Praça  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Fabrício Gonzalez Nogueira  
Universidade Federal do Ceará (UFC)

Dedico este trabalho à minha família...  
eles são minha razão de viver.

## AGRADECIMENTOS

À CNPq, pelo apoio financeiro com a manutenção da bolsa de auxílio.

Ao meu orientador, Prof. Dr. Paulo Cesar Marques de Carvalho, pela excelente orientação e acolhida desde o início da jornada do mestrado. Sua forma de acompanhar o desenvolvimento de meu trabalho contribuiu de maneira significativa na minha superação e formação acadêmica.

Ao meu co-orientador, Prof. Dr. Sandro César Silveira Jucá, pelas sugestões e acompanhamento em momentos importantes da pesquisa e pela experiência adquirida nas atividades conjuntas.

Aos professores participantes da banca examinadora Prof. Prof. Dr. Paulo Peixoto Praça e Prof. Dr. Fabrício Gonzalez Nogueira pelo tempo, pelas valiosas colaborações e sugestões que contribuíram significativamente na evolução deste trabalho.

Aos novos amigos, por sua amizade, pelas preciosas vivências e por seus ensinamentos. Especialmente pelas reflexões, críticas e sugestões recebidas ao longo da pesquisa, essenciais para o desenvolvimento deste trabalho.

A todos os Professores do Programa de Pós-Graduação em Engenharia Elétrica (PPGEE) da Universidade Federal do Ceará (UFC) pelos ensinamentos durante as disciplinas cursadas.

Ao Laboratório de Energias Alternativas (LEA) da UFC pela disponibilidade dos equipamentos e à Universidade Federal do Ceará por promover o incentivo à Pesquisa e à Pós-Graduação.

Finalmente, agradecer a minha família especialmente a minha mãe e a minha avó por ser parte essencial da minha vida e por tolerar todo o tempo que estivemos separadas; lutando pelo meu desenvolvimento acadêmico e profissional para garantir um futuro melhor. Agradeço também ao meu esposo por nos comprometer juntos nesta nova etapa das nossas vidas.

“Um problema sem solução, sem dúvida,  
é um problema mal colocado”.

Albert Einstein

## RESUMO

A presente dissertação tem como objetivo principal o desenvolvimento de um Sistema Embarcado Linux (ESL - *Embedded Linux System*) de monitoramento em Nuvem utilizando *software* livre. O bloco de aquisição de dados está composto por um Sistema Embarcado Conversor Analógico/ Digital (ADCES - *Analog/Digital Converter Embedded System*) e um ESL desenvolvido na placa *Raspberry Pi* que envia os dados coletados a uma interface entre a aquisição de dados e os serviços em Nuvem. Na presente pesquisa é utilizada uma planta de bombeamento fotovoltaico (FV) instalada no Laboratório de Energias Alternativas (LEA) localizado no Campus do PICI da Universidade Federal do Ceará (UFC) em Fortaleza - CE / Brasil. O trabalho apresenta uma análise comparativa entre o SAD desenvolvido e um sistema de supervisão e aquisição de dados comercial (SAD comercial). Comparando com as mesmas informações coletadas pelo SAD comercial, observa-se que os resultados do SAD desenvolvido apresentam um erro de medição em torno de 3%, proporcionando um grau de confiabilidade aceitável dos dados coletados. A pequena dimensão do dispositivo e sua ligação com o monitoramento em Nuvem e o uso de ferramentas com base em Internet das Coisas (*IoT*) são fatores determinantes para seu uso em diversas áreas, permitindo aumentar sua capacidade de coletar, analisar e distribuir dados em tempo real. A arquitetura de monitoramento *online* apresentada pode ser aplicada a projetos de microgeração descentralizada de energia elétrica a partir de diversas fontes renováveis de energia, colaborando com a tendência de diversificação da geração elétrica mundial e permite acrescentar funcionalidades e configurações de acordo com as necessidades dos usuários.

**Palavras-chave:** Aquisição de Dados, Energias Renováveis, Monitoramento Online.

## ABSTRACT

Commercial data acquisition systems (DAS) often have high costs and complex monitoring, and it is difficult to develop new features according to customer needs. In this sense, the main objective of this dissertation is the development of embedded Linux (Embedded Linux System) Cloud Monitoring System using free software. The data acquisition unit is composed of an Analog / Digital Converter Embedded System, an ESL developed on Raspberry Pi (Rpi) that sends the data collected from the sensors to an interface between the acquisition and Cloud services. In the present research, a photovoltaic (PV) pumping plant installed in the Alternative Energies Laboratory (LEA) located at the PICI Campus of the Federal University of Ceará (UFC) in Fortaleza - CE / Brazil is used. This work presents a comparative analysis between the developed DAS and a commercial data acquisition and supervision system (commercial DAS); with the purpose of studying the strengths and weaknesses of the new system compared to one already consolidated in the market. The measurements of the developed SAD were compared with the same information collected by the commercial DAS installed in the LEA; showing in all cases, a similar behavior between the curves of the parameters compared. The small size of the device and its connection to Cloud Monitoring and the use of Internet of Things (IoT) are determining factors for its use in several areas; allowing increasing our ability to collect, analyze and distribute data in real time. The proposed online monitoring architecture can be applied to decentralized electricity microgeneration projects from various renewable energy sources, contributing to the diversification tendency of the world electricity production.

**Keywords:** Data Acquisition, Renewable Energy, Online Monitoring

## LISTA DE FIGURA

Figura 1 - Diagrama de um sistema de aquisição de dados. ....	22
Figura 2 - Percentuais das características das bibliografias citadas. ....	28
Figura 3 - Diagrama esquemático de um sistema FV de abastecimento de água. ....	33
Figura 4 - Diagrama esquemático da planta de bombeamento FV.....	34
Figura 5 - Painel FV YL95P. ....	34
Figura 6 - Conjunto motobomba Shurflo 8000. ....	36
Figura 7 - Dimensões físicas das caixas FORTLEV 310 L. ....	37
Figura 8 - Divisor de tensão. ....	38
Figura 9 - Circuito divisor de tensão empregado como sensor de tensão. ....	38
Figura 10 - Sensor de corrente ACS712 e esquemático de ligação do sensor. ....	39
Figura 11 - Sensor PT-100 de 3 fios ....	40
Figura 12 - Sensor PT-100 e Transmissor NOVUS TxRail ....	40
Figura 13 - Conexão do sensor de temperatura PT-100. ....	41
Figura 14 - Conexão entre o transmissor e o ADC.....	42
Figura 15 - Conexão entre o transmissor e o sistema com resistor ....	43
Figura 16 - Pirômetro LP02. ....	43
Figura 17 - Placa amplificadora do sinal. ....	44
Figura 18 - Circuito divisor resistivo. ....	44
Figura 19 - Sensor DHT11 e aplicação do sensor. ....	45
Figura 20 - Sensor de fluxo volumétrico SU7000. ....	46
Figura 21 - Sensor de pressão combinada PN2024. ....	46
Figura 22 - Camadas do modelo de monitoramento em nuvem ....	50
Figura 23 - Ilustração do PaaS e do REMS proposto. ....	50
Figura 24 - Arquitetura do sistema embarcado proposto para monitoramento online. ....	51
Figura 25 - Principais características da placa Raspberry Pi 2 Modelo B ....	54
Figura 26 - Distribuição física dos pinos do PIC18F2550 ....	57
Figura 27 - Diagrama de blocos do ADC ....	58
Figura 28 - Processo de gravação do microcontrolador ....	59
Figura 29 - Conexão dos sensores analógicos à placa SanUSB. ....	59
Figura 30 - Conexão do sensor digital DHT11 à placa Rpi.....	61
Figura 31 - Fluxograma de tratamento dos dados dos sensores. ....	64
Figura 32 - Distribuição dos pinos GPIO do Raspberry Pi 2 Modelo B. ....	64
Figura 33 - Conexão serial entre a placa SanUSB (a) e o Raspberry Pi (b). ....	65

Figura 34 - Interface de comunicação para gravação no banco de dados. ....	69
Figura 35 - Fluxograma do programa para receber dados. ....	70
Figura 36 - Menu do sistema. ....	72
Figura 37 - Painel principal do sistema ....	73
Figura 38 - Exemplo de gráfico gerado pelo SAD ....	74
Figura 39 - Temperatura do painel FV e irradiância solar no dia 18/09/2016. ....	75
Figura 40 - Tensão, corrente e potência do painel FV no dia 18/09/2016. ....	75
Figura 41 - Tensão, corrente e potência do painel FV no dia 17/09/2016. ....	76
Figura 42 - Temperatura do painel FV e irradiância solar no dia 17/09/2016. ....	76
Figura 43 - Vazão e irradiância no dia 14/09/2016. ....	77
Figura 44 - Vazão e irradiância no dia 16/09/2016. ....	77
Figura 45 - Vazão, pressão e irradiância no dia 17/09/2016. ....	78
Figura 46 - Potência elétrica FV em função da irradiância no dia 19/09/2016. ....	78
Figura 47 - Potência máxima do módulo FV e P&O, CondInc, Sem MPPT ....	79
Figura 48 - Diagrama das energias calculadas no sistema. ....	81
Figura 49 - Sistema supervisor para coletar e armazenar dados. ....	83
Figura 50 - Tela do aplicativo desenvolvido para captura de dados ....	84
Figura 51 - Comparação da tensão do painel FV entre SAD e CLP (30/11/2016). ....	85
Figura 52 - Comparação da corrente gerada pelo painel FV entre SAD e CLP ....	85
Figura 53 - Comparação da irradiância solar disponível entre SAD e CLP ....	86
Figura 54 - Comparação da pressão entre SAD e CLP.....	87
Figura 55 - Comparação da vazão entre SAD e CLP.....	88
Figura 56 - Comparação da temperatura do painel FV entre SAD e CLP ....	88
Figura 57 - Internet das Coisas ....	90
Figura 58 - Elementos do ciclo de dados da Internet das Coisas ....	91
Figura 59 - Implementação do mecanismo Pull. ....	92
Figura 60 - Estrutura proposta para o monitoramento via Nuvem ....	93
Figura 61 - Opções para o acionamento da planta via Internet. ....	94
Figura 62 - Acionamento da motobomba (a) e bateria (b). ....	95
Figura 63 - Resultado da execução dos comandos no Rpi. ....	96

## LISTA DE TABELAS

Tabela 1 - Principais características das bibliografias citadas. ....	29
Tabela 2 - Especificações técnicas do painel YL95P. ....	35
Tabela 3 - Especificações do desempenho da motobomba SHURflo 8000-443-136.....	36
Tabela 4 - Sensores aceitos pelo transmissor .....	41
Tabela 5 - Características do PIC18F2550 .....	57
Tabela 6 – Conexão dos sensores nas entradas do PIC. ....	60
Tabela 7 - Formato de configuração do crontab. ....	63
Tabela 8 - Configuração do crontab usada na pesquisa. ....	63
Tabela 9 - Serviços por tipo de arquitetura do modelo de monitoramento em nuvem.....	66
Tabela 10 - Eficiência calculada pelo SAD desenvolvido .....	80
Tabela 11 - Energia utilizada, disponível e gerada pelo modulo FV .....	82
Tabela 12 - Análise comparativa dos SAD estudados. ....	90
Tabela 13 - Portas GPIO e ações a executar. ....	94

## LISTA DE ABREVIATURAS E SIGLAS

AD	Analogico Digital
ADCES	Analog/Digital Converter Embedded System
BDC	Backup Domain Controller
CGI	Common Gateway Interface
COM	Communication port in Windows
CPU	Central Processing Unit
EEPROM	Electrically Erasable Programmable Read Only Memory
ELS	Embedded Linux System
FER	Fontes de Energia Renovável
FTP	File Transfer Protocol
FV	Fotovoltaico
GCC	GNU Compiler Collection
GNU	GNU's Not Unix
GPIO	General-purpose input/output
GPRS	General Packet Radio Service
GPU	Graphics Processing Unit
GSM	Global System for Mobile Communication
HDMI	High Definition Multimedia Interface
HID	Human Interface Device
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IMAP	Internet Message Access Protocol
IoT	Internet of Things
LAN	Local Area network
LCD	Liquid Crystal Display
LDAP	Lightweight Directory Access Protocol
LED	Light emitter diode
LEA	Laboratório de Energias Alternativas
MIME	Multipurpose Internet Mail Extensions
NIST	National Institute of Standards and Technology
NNTP	Network New Transfer Protocol
PHP	Hypertext preprocessor

PIC	Programmable Interface Controller
PaaS	Platform as a Service
PDC	Primary Domain Controller
POP3	Post Office Protocol, version 3
RAM	Random Access Memory
REMS	Renewable Energy Monitoring System
RISC	Reduced Instruction Set Computer
ROM	Read Only Memory
Rpi	Raspberry Pi
SaaS	Software as a Service
SAD	Sistemas de Aquisição de Dados
SD	Secure Digital
SFCR	Sistema fotovoltaico conectado à rede elétrica
SMB	Server Message Block
SMS	Short text Message Service
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
SSH	Secure Shell
TCP	Transmission Control Protocol
TI	Tecnologia da informação
UFC	Universidade Federal do Ceará
USB	Universal Serial Bus
VNC	Virtual Network Computing

## SUMÁRIO

1	<b>INTRODUÇÃO</b> .....	17
1.1	<b>Plano da dissertação</b> .....	20
2	<b>REVISÃO BIBLIOGRÁFICA</b> .....	21
2.1	<b>Experiências no mundo</b> .....	21
2.2	<b>Experiências no Brasil</b> .....	24
2.3	<b>Análise da bibliografia pesquisada</b> .....	27
2.4	<b>Conclusões parciais</b> .....	31
3	<b>PLANTA DE BOMBEAMENTO FOTOVOLTAICO</b> .....	32
3.1	<b>Arquitetura da planta de bombeamento FV</b> .....	32
3.2	<b>Descrição dos materiais e equipamentos</b> .....	34
3.2.1	<i>Módulo fotovoltaico</i> .....	34
3.2.2	<i>Conjunto motobomba e reservatórios de água</i> .....	35
3.2.3	<i>Reservatórios de água</i> .....	37
3.2.4	<i>Sensores configurados</i> .....	37
3.3	<b>Parâmetros calculados</b> .....	46
3.3.1	<i>Parâmetros elétricos</i> .....	46
3.3.2	<i>Parâmetros hidráulicos</i> .....	47
3.3.3	<i>Eficiências do painel FV, do conjunto motobomba e global</i> .....	47
3.4	<b>Conclusões parciais</b> .....	48
4	<b>SISTEMA EMBARCADO PARA MONITORAMENTO ONLINE</b> .....	50
4.1	<b>Arquitetura geral do sistema</b> .....	50
4.2	<b>Aquisição de dados</b> .....	52
4.2.1	<i>Sistema Embarcado Conversor Analógico/Digital (ADCES)</i> .....	53
4.2.2	<i>Plataforma Raspberry Pi</i> .....	53
4.2.3	<i>Descrição da conexão entre a placa SanUSB e o Raspberry Pi</i> .....	64
4.3	<b>Envio e armazenamento em nuvem</b> .....	65
4.3.1	<i>Servidor Web</i> .....	66
4.3.2	<i>Banco de dados</i> .....	68
4.3.3	<i>Gravação em nuvem dos componentes do sistema</i> .....	69
4.4	<b>Conclusões parciais</b> .....	70
5	<b>APLICAÇÃO E RESULTADOS</b> .....	71
5.1	<b>Princípio de operação do sistema</b> .....	71

5.2	<b>Interface gráfica</b> .....	72
5.3	<b>Principais resultados</b> .....	72
5.3.1	<i>Comportamento da temperatura do painel FV em relação à irradiância</i> .....	74
5.3.2	<i>Relação entre vazão e irradiação solar</i> .....	76
5.3.3	<i>Relação entre vazão, pressão e irradiância</i> .....	77
5.3.4	<i>Relação entre potência elétrica FV e irradiância</i> .....	78
5.3.5	<i>Eficiência do painel FV, do conjunto motobomba e do sistema geral</i> .....	79
5.4	<b>Comparação entre medições</b> .....	82
5.4.1	<i>Análise dos sistemas de aquisição comparados</i> .....	89
5.5	<b>Interconexão da planta de bombeamento fotovoltaico à rede IoT</b> .....	90
5.6	<b>Conclusões parciais</b> .....	97
6	<b>CONCLUSÃO</b> .....	98
	<b>Sugestões para trabalhos futuros</b> .....	99
	<b>REFERÊNCIAS</b> .....	100
	<b>APÊNDICES</b> .....	108

## 1 INTRODUÇÃO

O desenvolvimento tecnológico que tem ocorrido nos últimos anos, especialmente na área da eletrônica e da computação, fez com que empresas, centros de investigação e universidades desejem ter equipamentos e ferramentas com melhores características e funções com custos relativamente baixos. Na década de 90, as redes de comunicações mudaram a forma como as pessoas e organizações trocavam informações e organizavam as suas atividades. Os avanços tecnológicos tornaram os mecanismos distribuídos realidade: pequenos, baratos e de baixo consumo de energia, que são também capazes de adquirir informações assim como de se comunicar de forma sem fio. A disponibilidade de microsensores e comunicações sem fio fornece as ferramentas necessárias para implementar redes de sensores/atuadores úteis em uma ampla gama de aplicações.

A gestão de rede de sensores é o problema fundamental para as aplicações de monitoramento e controle. A crescente integração de transdutores de inteligência ou sensores/atuadores levou ao desenvolvimento de transdutores de informação inteligentes. A diversidade do mercado de transdutores resultou em uma abundância de soluções de baixo custo para redes de aquisição de dados, que são projetados para aplicações específicas (LABRADA; HERNÁNDEZ, 2007). Geralmente, ferramentas e equipamentos concebidos para aplicações específicas têm como objetivo principal o baixo custo e oferecem melhor relação custo-benefício em comparação com sistemas e tecnologias proprietárias, além de oferecer um ambiente de fácil assimilação e operação (JUCÁ, 2014).

Algumas das áreas em que essas redes são mais usadas são: a indústria automotiva, automação de fábricas em redes de controle industriais, automação de edifícios e escritórios, monitoramento, aquisição e transmissão de dados do funcionamento de sistemas de geração de energia elétrica, entre outros, mas seu potencial é tão grande que a sua aplicação pode ser limitada apenas pela imaginação humana. Assim, os recentes esforços visam à interoperabilidade em uma ampla gama de aplicações, tentando concentrar-se na definição, funcionalidade e protocolos de comunicação padrão de sensores inteligentes, criação de normas, cujo objetivo principal é definir uma arquitetura que permite aos transdutores conectar-se a qualquer rede de controle de modo *plug and play* facilitando sua identificação e configuração na rede (DEVARAJAN, 2001).

O monitoramento dos processos de geração de energia elétrica é uma tarefa essencial caso se queira ter resultados eficientes. No entanto, as ferramentas com este

propósito específico são desenvolvidas com tecnologias proprietárias e importadas, fazendo aumentar os custos e inviabilizando a implantação, a difusão e o aprendizado. Para atender essas limitações, têm sido desenvolvidas pesquisas que aproveitam as características dos sistemas microcontrolados e das plataformas de código aberto tais como BRITO *et al.*, (2014), (HALMEMAN, 2014), JUCÁ *et al.* (2011), JUCÁ *et al.*, (2008), MAHJOUBI *et al.* (2011). Isto proporciona maior flexibilidade e autonomia nas futuras soluções de monitoramento e aquisição de dados.

Um microcontrolador é um circuito integrado que contém no interior uma unidade de processamento central (*CPU - Central Processing Unit*), unidades de memória (*RAM - Random Access Memory* e *ROM - Read Only Memory*), portas de entrada/saída (*Input/Output*) e periféricos tais como canal USB e conversores Analógico/Digital (A/D), entre outros. (IOVINE, 2000) (KEITHLEY INSTRUMENTS, 2001). Estas partes são interligadas dentro do microcontrolador e juntos formam o que é conhecido como um microcomputador. Pode-se dizer que um microcontrolador é um microcomputador completa encapsulado num circuito integrado. O baixo custo, variedade e capacidade dos microcontroladores para armazenar e executar programas, além da leitura e comunicação dos sensores, torna-os muito versáteis. A capacidade dos microcontroladores para executar funções matemáticas e lógicas permite o desenvolvimento de uma lógica sofisticada em equipamentos eletrônicos. Alguns programas podem fazer os microcontroladores se comportarem como um circuito neural, uma vez que estes são responsáveis pela lógica na maioria dos dispositivos inteligentes desenvolvidos. O uso de microcontroladores em SAD (Sistemas de Aquisição de Dados) adiciona um grande potencial, variedade de opções e possibilidades de controle por isso seu uso é essencial a fim de manter os níveis de competência e de combinar as vantagens proporcionadas por estes sistemas (IOVINE, 2000).

O estudo realizado pela literatura consultada permitiu perceber que o uso de redes de comunicação sem fio para SAD é uma opção atraente que fornece um modelo de conectividade sem fio reconfigurável. Recentes pesquisas de redes de sensores sem fio tem se concentrado na concepção de protocolos de eficiência energética para a comunicação entre nós de sensores, onde milhares desses nós participam na comunicação. Outra abordagem a este tipo de rede é integrar vários protocolos de comunicação de tecnologia sem fio. (DEVARAJAN, 2001). A norma 802.11, também conhecido como *WiFi*, é a mais amplamente usada em redes de computadores em todo o mundo; o alcance, baixo consumo e velocidade de transmissão a torna ideal para o estabelecimento de comunicação sem fio ponto

a ponto e entre nós e redes distantes (IEEE, 2015). Este protocolo de comunicação sem fio será usado neste projeto para enviar os dados adquiridos para a nuvem onde vai estar tanto o servidor *Web* como o banco de dados.

É definido como problema de estudo o monitoramento *online* aplicado a plantas de geração descentralizada de energia elétrica utilizando fontes renováveis. A pesquisa é desenvolvida no Laboratório de Energias Alternativas da Universidade Federal do Ceará (LEA-UFC). Para resolver o problema apresentado acima é definido como objetivo geral da presente dissertação: Desenvolver um sistema embarcado Linux para o monitoramento em nuvem de uma planta de geração descentralizada de eletricidade aplicada ao bombeamento fotovoltaico.

Para cumprir com o objetivo geral acima definido, os seguintes objetivos específicos são propostos:

1. Realizar uma revisão bibliográfica de pesquisas e aplicações de SAD a nível nacional e internacional a título de justificar a investigação.
2. Descrever a planta de bombeamento fotovoltaica instalada no LEA-UFC utilizada para o desenvolvimento da pesquisa.
3. Apresentar uma descrição do SAD proposto e dos componentes que são parte da solução desenvolvida.
4. Oferecer os principais resultados e aplicações do sistema proposto comparado com o SAD comercial instalado no LEA para determinar possíveis falhas e corrigi-las.

Para o desenvolvimento do trabalho e a fim de cumprir o objetivo geral estabelecido assim como os objetivos específicos desta pesquisa, são propostas as seguintes tarefas de investigação:

1. Identificar tendências e tecnologias necessárias para o desenvolvimento do projeto de pesquisa de acordo com as restrições das instalações utilizadas.
2. Investigar e comparar padrões e plataformas de *software* livre para o desenvolvimento de aplicativos de supervisão e aquisição de dados.
3. Estudar formas de acesso remoto às plantas de geração descentralizada, permitindo uma operação à distância do sistema com redução de custo de deslocamentos.
4. Registrar dados dos sensores analógicos e digitais conectados à planta a fim de acompanhar o desempenho do sistema em tempo real.

5. Projetar o modelo de dados que vai armazenar as informações dos sensores ligados em vários pontos do sistema com a finalidade de acompanhamento detalhado do comportamento do sistema.

6. Projetar e implementar a solução tendo em conta a infraestrutura existente, composta de um sistema microcontrolado de aquisição de dados com transmissão via *Wifi* e gravação *online* em nuvem de um banco de dados para consulta das informações do sistema.

7. Fundamentar os aspectos relevantes da pesquisa e do trabalho realizado, a modo de facilitar a manutenção, suporte e implementação de futuras atualizações da solução desenvolvida.

Espera-se que este esforço seja útil para fornecer uma visão geral dos sistemas embarcados de aquisição de dados relacionados com as energias renováveis. Deve-se mencionar ainda que um dos focos deste trabalho é criar melhores condições de aproveitamento da riqueza e do potencial dos recursos renováveis em benefício das gerações futuras.

### **1.1 Plano da dissertação**

O plano da dissertação consiste em um resumo, uma introdução que apresenta os objetivos e a metodologia do trabalho, bem como o plano de desenvolvimento da dissertação; e vários capítulos como é descrito a seguir:

Capítulo 1: Apresenta a introdução, as motivações da pesquisa e os objetivos do trabalho.

Capítulo 2: Contém uma revisão bibliográfica mundial e nacional de SAD aplicados a fontes alternativas.

Capítulo 3: Descreve a planta de bombeamento de água abastecida por um módulo fotovoltaico utilizada na pesquisa assim como a descrição dos diferentes sensores configurados. Contém a modelagem matemática de parâmetros envolvidos em sistemas de bombeamento de água.

Capítulo 4: Aborda o desenvolvimento do sistema embarcado Linux proposto utilizado para monitorar os dados dos sensores configurados no LEA-UFC.

Capítulo 5: Oferece os principais resultados e aplicações do sistema proposto assim como as comparações feitas com o SAD comercial instalado no LEA para determinar possíveis falhas e corrigi-las.

Capítulo 6: Apresenta as conclusões finais da pesquisa das diferentes etapas de realização do projeto assim como as dificuldades encontradas. Aborda as perspectivas de utilização desta dissertação como uma base para futuras pesquisas.

## 2 REVISÃO BIBLIOGRÁFICA

O presente capítulo apresenta uma revisão bibliográfica sobre a aquisição de dados no Brasil e no mundo, incluindo uma abordagem das aplicações dos SAD em plantas de bombeamento de água que utilizam tecnologia solar fotovoltaica (FV) como alimentação. Apesar da revisão bibliográfica não abordar todos os trabalhos e pesquisas desenvolvidos, apresenta-se um estudo dos principais trabalhos sobre o tema, auxiliando na compreensão dos objetivos propostos.

### 2.1 Experiências no mundo

Embora conceitos como aquisição, medição de dados e o tratamento deles possam ser conceitos difíceis de definir, a maioria dos usuários, engenheiros e cientistas concordam que há vários elementos comuns (KEITHLEY INSTRUMENTS, 2001). A Figura 1 mostra a estrutura básica de um SAD composto pelos elementos básicos. Uma parte fundamental de qualquer SAD são os sensores, elementos encarregados de adquirir informações das variáveis a serem medidas, como ilustrado na Figura 1a. Os sensores são dispositivos compostos pelo elemento físico sensível e pelo transdutor que traduz as manifestações físicas em sinais elétricos (MEASUREMENT COMPUTING, 2012). O objetivo do condicionamento de sinal é gerar, a partir das informações obtidas pelos sensores, um sinal que seja aceitável para a placa de aquisição de dados, conforme Figura 1b. Este processo melhora a qualidade do sinal gerado antes de serem convertidos em sinais digitais. Exemplos de condicionamento de sinal incluem: transformação, amplificação, filtragem, atenuação, excitação, linearização. Outra etapa nos SAD é a aquisição de dados através das placas responsáveis pela conversão dos sinais analógicos para digital e comunicação com o computador, conforme Figura 1c (MORA, 2011). Finalmente é usado um computador para programar o sistema e manipular ou armazenar os dados coletados, conforme Figura 1d. Este termo generaliza qualquer computador que executa a ferramenta de monitoramento para obter relatórios e gráficos, geralmente em tempo real (KEITHLEY INSTRUMENTS, 2001). Embora os SAD tenham evoluído ao longo do tempo desde gravadores eletromecânicos até sistemas eletrônicos capazes de medir centenas de variáveis simultaneamente; todos seguem a mesma estrutura básica apresentada na Figura 1. A maioria deles são compostos por um conjunto de sensores para medir as diferentes variáveis. Depois é realizado um processo de conversão e adaptação da magnitude física num sinal eletrônico para sua leitura digital para fazer a aquisição de dados, geralmente através de um computador. Com a evolução dos dispositivos capazes de

converter grandezas físicas mensuráveis em sinais elétricos, estes sistemas começaram a ser aplicados às mais diversas áreas de conhecimento.

Figura 1 - Diagrama de um sistema de aquisição de dados.



Fonte: Elaborado pelo autor.

No mundo todo existem muitas pesquisas relacionadas com o presente tema de estudo que podem servir como referências. Com relação ao bombeamento de água, a concepção e desenvolvimento de um SAD sem fio baseado no microcontrolador PIC16F877 e um estudo sobre a viabilidade de diferentes metodologias existentes relacionadas à aquisição de dados a partir de sistemas de bombeamento de água acionados por módulos FV é apresentado em MAHJOUBI *et al.* (2011). O sistema é usado para coletar, armazenar e analisar informações de várias estações de bombeamento de água em áreas remotas na região árida do sul da Tunísia. Estas informações podem ser visualizadas através da Internet. Neste estudo, foram analisadas várias técnicas de transmissão de dados existentes, tais como transmissão via satélite, rádio, GSM (*Global System for Mobile Communication*) e GPRS (*General Packet Radio Service*). As comunicações remotas são baseadas na rede GSM e, em especial, no serviço de SMS (*Short text Message Service*).

Um projeto de um SAD na Argélia pode ser encontrado em BENGHANEM *et al.* (1999). O sistema oferece a análise do desempenho de um sistema FV de bombeamento de água que pode ser acessado desde o servidor central. O sistema apresentado pode se adaptar a qualquer tipo de aquisição de dados e oferece a vantagem de ser autônomo. A arquitetura do sistema está estruturada em torno do microprocessador MC68B09. Neste trabalho é usado um conversor analógico/digital ADC0804 junto com o barramento de dados. O tratamento dos dados é feito em tempo real e são transferidos a um computador pessoal através de uma porta serial bidirecional RS232C.

Um novo conceito de sistema embarcado servidor/cliente incorporado às fontes de energia renováveis (FER) para o monitoramento de painéis FV e baterias que estão se carregando através do sistema solar, pode ser visto em (FATING; NAGRARE, 2013). Neste

artigo, o servidor embarcado é desenvolvido com o controlador ARM9 junto com o sistema operacional *Windows CE* ®. O cliente de baixo custo é projetado usando o microcontrolador ATmega32 com conexão à rede de área local (LAN - *Local Area Network*). Foram desenvolvidas aplicações na parte do servidor para monitorar e controlar a operação na parte do cliente, podendo ser acessadas via Internet. O sistema desenvolvido oferece as seguintes funcionalidades: monitoramento do status da bateria que está sendo carregada, monitoramento e controle de estado das cargas como ventiladores e lâmpadas e o intercâmbio de informações com o servidor por meio da LAN.

Em KRISHNA *et al.*, (2012) é descrito o desenvolvimento de um SAD para o monitoramento de sistemas de geração FV. O sistema está baseado na placa de aquisição de dados Campbell CR3000 e a ferramenta de programação gráfica PC 400. O sistema proposto permite medir dados meteorológicos como temperatura e umidade, tanto como parâmetros elétricos como tensão e corrente FV. A arquitetura está estruturada em torno do microprocessador MC57B98. No sistema é medida a temperatura ambiente do dia, a irradiação solar diária e horária, a taxa de vazão e o nível dos tanques.

O desenvolvimento de um SAD para o monitoramento e controle de plantas remotas de FER é apresentado em (KALAITZAKIS; KOUTROULIS; VLACHOS, 2003). O sistema é baseado na arquitetura cliente/servidor que não requer conexão física entre os sistemas monitorados e o servidor. Essa opção é essencial devido a que as plantas de energia renováveis são geralmente instaladas em áreas pouco acessíveis. O usuário pode acessar os dados medidos através da Internet. Neste trabalho foi proposto que o *software* executado deve ser capaz de rodar em qualquer máquina, independentemente do sistema operacional e da CPU utilizada, garantindo a independência de plataforma. Um sistema de medição distribuído totalmente funcional baseado na arquitetura descrita é implementado no campus da Universidade Técnica de Creta, Grécia. A aplicação principal desta pesquisa é o monitoramento de parâmetros meteorológicos e elétricos de um gerador eólico, dois conjuntos fotovoltaicos, um sistema de bateria, carregadores de bateria e inversores.

Um sistema para a aquisição de dados meteorológicos desenvolvido em Argel é apresentado em ADANE *et al.*, (2014) apresentam. Este sistema permite coletar informação sobre temperatura do ar, pressão atmosférica e umidade relativa sob várias condições climáticas. Foi usado o compilador PCWH e a linguagem C para realizar a programação do sistema. O sistema permite obter medições a cada hora da temperatura do ar, pressão atmosférica e umidade relativa. O microcontrolador PIC16F8877A assegura a gestão de todo

o sistema. Os dados coletados são exibidos numa tela LCD (*Liquid Crystal Display*) e são transferidos para um computador pessoal através da interface serial RS232, onde são armazenados e processados. O sistema é alimentado por uma fonte de energia autônoma FV e está equipado com um sistema de transmissão de dado.

O projeto, desenho e a implementação de um SAD com comunicação sem fio para o monitoramento remoto de variáveis ambientais pode ser encontrado em ROQUE *et al.*, (2013) pode ser encontrado. O sistema usa um microcontrolador PIC16F877A e um computador com um módulo de transmissão/recepção incorporado baseado em tecnologia *ZigBee*. O funcionamento dos programas é verificado em ambientes de simulação. O sistema pode ser adaptado de forma relativamente fácil para atender vários tipos de sensores com saída de tensão com ligeiras alterações no programa em execução no microcontrolador e adaptando o programa LabVIEW em execução no PC com as características de cada sensor. Isto torna-o mais flexível, sem necessidade de alterar o núcleo do bloco de aquisição. Podem ser atendidos remotamente até 8 sensores com diferentes tensões de saída e, portanto, controlar esse número de diferentes variáveis de interesse, de forma simples e a um custo mais baixo do que outras tecnologias.

## 2.2 Experiências no Brasil

Existem muitos projetos de geração FV no Brasil para o suprimento de eletricidade em comunidades rurais e/ou isoladas no Norte e Nordeste do Brasil. Esses projetos atuam basicamente com quatro tipos de sistemas autônomos:

- a) Bombeamento de água para abastecimento doméstico, irrigação e piscicultura
- b) Iluminação pública
- c) Sistemas de uso coletivo, tais como eletrificação de escolas, postos de saúde e centros comunitários
- d) Atendimento a casas, tais como as estações de telefonia e monitoramento remoto, a eletrificação de cercas, a produção de gelo e a dessalinização de água (ANEEL, 2005).

Nesta seção são apresentados alguns exemplos dos SAD desenvolvidos para o monitoramento destes projetos.

O desenvolvimento de um sistema de monitoramento para avaliação de desempenho de um sistema FV conectado à rede de baixa tensão é apresentado em (CARVALHO *et al.*, 2016). Esta pesquisa foi desenvolvida pelo Grupo de Estudos e

Desenvolvimento de Alternativas Energéticas (GEDAE) da Universidade Federal do Pará (UFPA). O sistema monitorado é composto por 6 módulos FV de 245 Wp em série, modelo YL245P-29b, constituindo assim um gerador fotovoltaico de 1,47 kWp de potência. O inversor utilizado no sistema é o inversor *grid tie* SB 1200, da marca SMA, que apresenta uma potência nominal de 1200 W. Foi utilizado Arduino Mega 2560 como unidade central de aquisição e tratamento dos dados. O sistema permite a medição de tensão e corrente do gerador FV, temperatura na parte posterior do módulo FV e irradiância solar. As informações armazenadas em um cartão de memória SD (*Secure Digital*) são visualizadas de forma local em tempo real através dos displays LCD instalados.

O desenvolvimento e implementação de um Sistema FV conectado à rede elétrica (SFCR) utilizando Arduino como plataforma de desenvolvimento pode ser encontrado em DIAS *et al.*, (2016). Este trabalho foi desenvolvido no Laboratório de Energias Renováveis da Universidade do Vale do Rio dos Sinos, Unisinos, situado na cidade de São Leopoldo. O sistema está composto por um painel FV de 1080 Wp, de silício policristalino, um inversor CC/CA conectado à rede elétrica, sensores de corrente e tensão, assim como sensores de temperatura e irradiância. O sistema permite a monitorização de tensões e correntes, contínuas e alternadas, temperatura da célula e ambiente e irradiância. As aquisições feitas a cada minuto são mostradas em tempo real em um display LCD e armazenadas em um cartão de memória SD. Os dados coletados são exportados e tabulados no *software Microsoft Excel* para fazer gráficos característicos de geração de energia com relação às condições climáticas específicas.

Um sistema de monitoramento e controle de consumo de energia elétrica para uma residência é proposto em (BARRETO, 2015). O projeto propõe o monitoramento dos níveis de consumo de energia elétrica em duas cargas monofásicas comumente utilizadas em uma residência, relacionadas à iluminação e condicionamento ambiental. Este trabalho apresenta uma forma de controle do consumo de energia de duas cargas monofásicas referentes à iluminação e condicionamento ambiental. É usado o modelo Arduino UNO R3 da plataforma Arduino para medir os valores de consumo energético. A apresentação dos dados é feita através de uma interface gráfica implementada utilizando a plataforma Raspberry Pi (*Rpi*). O código fonte do sistema principal é desenvolvido em Python, através da IDE Geany no *Rpi*. A comunicação entre os leitores de consumo de energia e o sistema principal é realizada pela porta serial USB de ambas as plataformas (Arduino e Raspberry Pi).

Um sistema remoto de aquisição e monitoramento de variáveis ambientais para aplicações de previsão meteorológica desenvolvido na Universidade Federal do Ceará (UFC) é apresentado em (MENEZES, 2015). O sistema é composto por estações de campo, estações de usuários e um computador dedicado fazendo a função de servidor Web. As estações de campo realizam medições e transmitem as informações para o servidor Web. O projeto dispõe de um microcontrolador PIC18F4520 para coletar dados dos sensores de temperatura, volume de chuva e velocidade e direção do vento e salvar as medições em um banco de memória local. A informação é publicada na Internet para um servidor de dados junto com um sistema de banco de dados.

Um novo conceito de SAD de baixo custo aplicado às plantas de energias renováveis descentralizadas, com base em *software* livre e uma interface USB é apresentado em JUCÁ *et al.*, (2011) apresentam. No sistema proposto, as variáveis são diretamente armazenadas na EEPROM (*Electrically Erasable Programmable Read Only Memory*), fazendo com que o sistema de monitoramento seja mais barato. O SAD apresenta também uma interface de comunicação TTL/RS-232 que pode ser usada como uma interface para modems GSM/GPRS e um transceptor TTL/RS-485 para uso em dispositivos Modbus. Os dados armazenados podem ser lidos localmente através da interface USB ou podem ser transmitidos via sem fio a servidores através de um modem GSM/GPRS ou para uma rede Modbus usando o transceptor TTL/RS485. O sistema desenvolvido foi utilizado para medir dados de velocidade do vento e dados de pressão e tensão de uma unidade de bombeamento de água alimentada por módulos FV.

Um SAD desenvolvido para monitoramento de um SFCR integrado à edificação é apresentado em MANITO *et al.*, (2010). Os dados podem ser visualizados na Internet por meio de um programa computacional desenvolvido em ambiente LABVIEW depois que estes sejam tratados, armazenados e publicados. Neste trabalho foram desenvolvidos *software* e *hardware* que permitem a obtenção de parâmetros elétricos como tensão e corrente CC e CA, além da aquisição de dados climáticos, temperatura de costa da célula FV e irradiância, além de estar preparado para a introdução de diversos outros parâmetros, caso necessário. Os dados publicados na Internet podem ser acessados através de usuário e senha por meio da comunicação com um servidor de FTP (*File Transfer Protocol*).

Um sistema microcontrolado de baixo custo para aquisição de dados e transmissão via GSM dos valores dos sensores de um sistema de energia renovável é apresentado em NETO *et al.*, (2010). O resgate dos dados armazenados na EEPROM externa

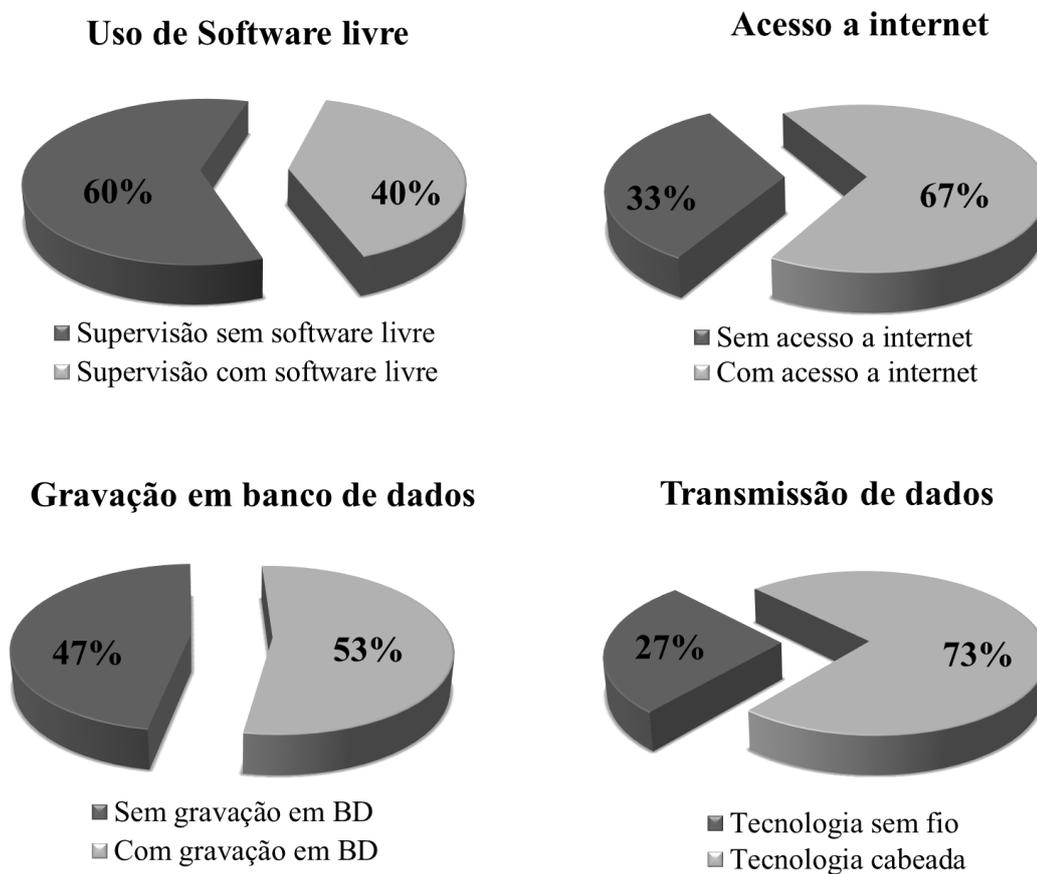
pode ser feito por qualquer *software* com interface serial através da emulação de um canal serial virtual via USB. Os dados armazenados são coletados no local por um computador através de um programa supervisorio via USB ou transmitidos via GSM utilizando a tecnologia GPRS para que os dados sejam visualizados em qualquer lugar do mundo. Foi usada a linguagem C para a programação do sistema de armazenamento de dados no microcontrolador e a interface serial com o modem. A programação interna do modem GSM e sua interface com um banco de dados foram feitas usando as linguagens Java e PHP.

O desenvolvimento de um sistema para monitoramento da produção e consumo de energia em centrais de microgeração fotovoltaicas pode ser encontrado em (HALMEMAN, 2014). O trabalho foi desenvolvido na central de microgeração distribuída fotovoltaica (MGD-PV) do Sítio Modelo da Fazenda Lageado e no Laboratório de Energias Renováveis do Departamento de Engenharia Rural, na Faculdade de Ciências Agrônomicas da UNESP, campus de Botucatu. O MGD-PV está composto por módulo FV, controlador de carga, inversores, baterias e bombas de água. O arranjo fotovoltaico projetado é um sistema autônomo, com potência de 1536 Wp. O sistema permite a medição da tensão e corrente por meio de uma arquitetura de *software* e *hardware* livres. Como protocolo de comunicação sem fio é usado ZigBee. A plataforma Arduino é utilizada para a coleta e armazenamento local dos dados e o envio para a Internet.

### **2.3 Análise da bibliografia pesquisada**

Na Tabela 1 é mostrado um resumo com as principais características dos SAD citados neste capítulo. A Figura 2 apresenta os percentuais das participações das características dos SAD referenciados. Como pode ser observado, apenas 40 % dos SAD citados usam *software* livre para o desenvolvimento da sua interface visual, enquanto 60 % fazem uso de *softwares* proprietários ou pagos para realizar a mesma tarefa. Em relação ao acesso à Internet, 67 % dos trabalhos estudados fazem menção desta característica, desta maneira 33 % trabalham localmente, o que resulta na utilização de computadores dedicados. A utilização de banco de dados está presente em 53 % das referências citadas enquanto o 47 % dos trabalhos usam outros métodos de armazenamento. Finalmente, a transmissão usando tecnologias sem fio está presente em 27 % das referências estudadas e 73 % que faz uso de tecnologias cabeadas como conexão serial RS-232 ou interface Ethernet.

Figura 2 - Percentuais das características das bibliografias citadas.



Fonte: Elaborado pelo autor.

Tabela 1 - Principais características das bibliografias citadas.

<b>Sistema</b>	<b>Aquisição</b>		<b>Transmissão</b>			<b>Supervisão</b>	<b>Aplicação</b>
<b>Autor</b>	<b>Conexão ao PC</b>	<b>Armazenamento permanente de dados</b>	<b>Acesso à internet</b>	<b>Gravação em banco de dados</b>	<b>Tecnologia de transmissão de dados</b>	<b>Interface visual</b>	
MAHJOUBI <i>et al.</i> , (2011)	TTL/RS-232	EEPROM externa	Sim	Sim	Conexão sem fio (GSM)	WinDev10	Bombeamento de água
BENGHANEM <i>et al.</i> , (1999)	RS-232C	RAM/ EPROM	Sim	Sim	Conexão Direta/Serial	Não definido	Bombeamento de água
(FATING; NAGRARE, 2013)	Interface Ethernet	Não definido	Sim	Não definido	Interface Ethernet	C# no Visual Studio 2008	Monitoramento dos parâmetros do painel fotovoltaico e do estado da bateria que está se recarregando
KRISHNA <i>et al.</i> , (2012)	TTL/RS-232	DAS EEPROM	Sim	Sim	Conexão Direta/Serial	Java	Geração de Energia Solar Fotovoltaica
KALAITZAKIS <i>et al.</i> , (2003)	Rádio Frequência/RS-232	Buffer	Sim	Não	Interface Ethernet	Java	Monitoramento de parâmetros meteorológicos e elétricos
JUCÁ <i>et al.</i> , (2011)	TTL/RS-232 TTL/RS-485 Interface USB	DAS EEPROM	Sim	Sim	Interface USB	Não definido	Monitoramento de velocidade do vento e pressão e tensão de uma unidade de bombeamento fotovoltaico
MANITO <i>et al.</i> , (2010)	Não definido	Não definido	Sim	Sim	Não definido	LABVIEW	Monitoramento de um sistema fotovoltaico conectado à rede

(BARRETO, 2015)	Serial USB	EEPROM	Não	Não	Serial USB	Python	Monitoramento do consumo elétrico de uma residência
(MENEZES, 2015)	Não definido	EEPROM	Sim	Sim	Conexão sem fio (GSM/GPRS)	PHP	Monitoramento de variáveis ambientais para previsão meteorológica
NETO <i>et al.</i> (2010)	Interface USB	EEPROM externa	Sim	Sim	Conexão sem fio (GSM/GPRS)	PHP/Java	Monitoramento de um sistema de bombeamento fotovoltaico de água
ADANE <i>et al.</i> , (2014)	TTL/RS-232	Não definido	Não	Não	Interface USB TTL/RS-232	Tela LCD com Delphi	Monitoramento de parâmetros meteorológicos
ROQUE <i>et al.</i> (2013)	TTL/RS-232	Não definido	Não	Não	Interface USB TTL/RS-232	LABVIEW	Monitoramento de variáveis ambientais
(HALMEMAN, 2014)	XBee	Memória SD	Sim	Sim	XBee	PHP	Monitoramento da produção e consumo de energia em centrais de microgeração fotovoltaicas
CARVALHO <i>et al.</i> , (2016)	Não definido	Memória SD	Não	Não	Não definido	Tela LCD	Monitoramento de um sistema FV conectado à rede
DIAS <i>et al.</i> (2016)	Não definido	Memória SD	Não	Não	Não definido	Tela LCD	Monitoramento de um sistema FV conectado à rede

Fonte: Elaborado pelo autor.

## 2.4 Conclusões parciais

Este capítulo aborda de forma geral os conceitos básicos relacionados com a obtenção, medição e tratamento de dados para o desenvolvimento de SAD, servindo de base para selecionar os diferentes componentes que constituem o sistema de monitoramento *online* proposto. Para atingir os objetivos do trabalho, são utilizadas as características mais importantes apresentadas nas referências citadas, tais como o acesso à Internet e a gravação em banco de dados, presente em elevada percentagem da bibliografia estudada, 69% e 54%, respectivamente.

Como meio de transmissão é utilizado o protocolo de comunicação sem fio dado que não foram totalmente exploradas suas funcionalidades, estando presente em apenas 27% das referências estudadas. Foi verificado que em nenhuma das referências citadas foi utilizado o conceito de monitoramento em nuvem para a aquisição de dados. Dessa forma, como contribuição do presente trabalho, é proposto um sistema de aquisição sem fio, com armazenamento de dados na Internet junto com um site hospedado em nuvem para a visualização de gráficos e *logs* gravados.

### **3 PLANTA DE BOMBEAMENTO FOTOVOLTAICO**

Este capítulo visa fornecer uma descrição da planta de bombeamento de água alimentada por módulos FV utilizada na pesquisa bem como a descrição dos diferentes sensores configurados. Também é abordada a modelagem matemática de parâmetros envolvidos em sistemas de bombeamento FV.

#### **3.1 Arquitetura da planta de bombeamento FV**

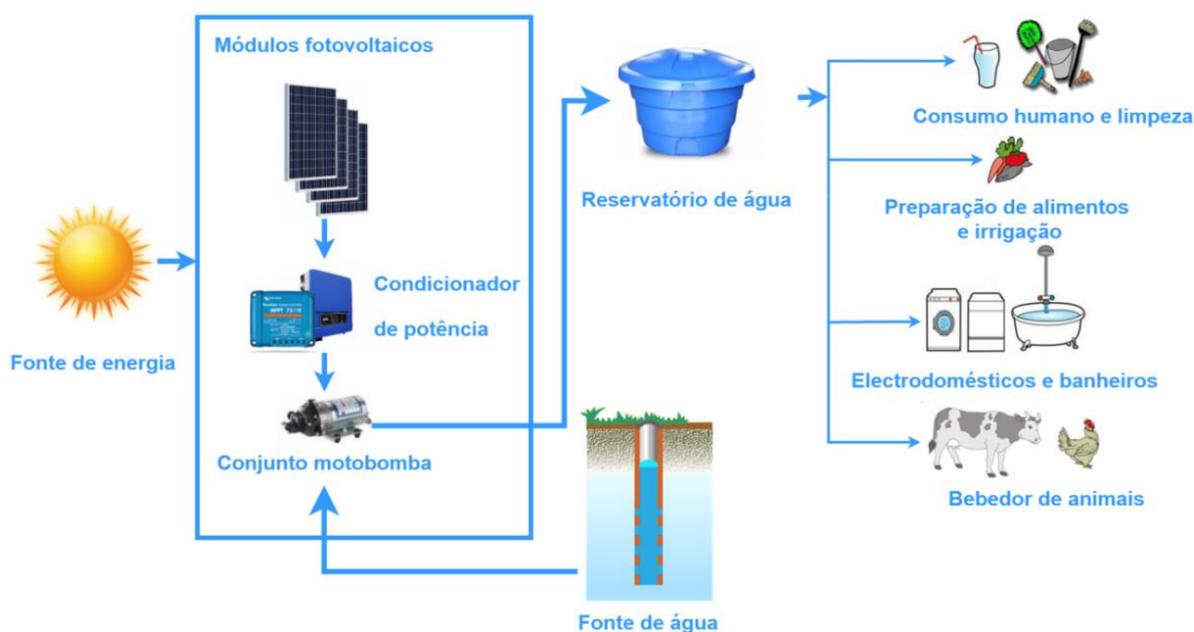
Uma das aplicações da energia solar é o bombeamento de água. A água, como a eletricidade, é um fator de saúde e de progresso para as comunidades, principalmente aquelas situadas em locais remotos. O uso de sistemas de bombeamento acionados com módulos fotovoltaicos é, hoje, uma realidade. Existem milhares de sistemas em funcionamento nas mais remotas regiões do mundo. Esses sistemas são eficientes, confiáveis, necessitam de pouca manutenção e resolveram o problema de abastecimento de água dessas comunidades com um custo relativamente baixo (ALVARENGA, 2015).

Um sistema de bombeamento FV geralmente é composto por um gerador FV, mecanismo de condicionamento de energia (inversor), conjunto motobomba e equipamentos complementares, como hidrômetro e sensor de nível (FEDRIZZI; SAUER, 2002), conforme a Figura 3. Para o desenvolvimento da presente pesquisa foi usada uma planta de bombeamento FV instalada no Laboratório de Energias Alternativas (LEA). O LEA está localizado no Campus do PICI da Universidade Federal do Ceará (UFC) em Fortaleza - CE / Brasil, com coordenadas UTM de latitude: 3° 44' 15" S e longitude: 38° 34' 23" W. Como regra geral, não são utilizadas baterias, pois os reservatórios de água desempenham a função de armazenamento (FEDRIZZI, 2003). Um aspecto muito favorável destes sistemas refere-se à possibilidade de armazenamento da água bombeada em vez do armazenamento da eletricidade produzida pelos módulos FV. Isso diminui a importância de um grande fator limitante do uso da energia solar que é sua variabilidade no tempo e a necessidade de sistemas de armazenamento para tê-la de forma contínua. Os reservatórios de água substituem as baterias elétricas a um custo muito menor (ALVARENGA, 2015). Como os sistemas de bombeamento FV sem baterias não fornecem água quando o sol não brilha com a intensidade necessária, é aconselhável ter reservatórios de armazenamento. Nos períodos noturnos ou quando não há insolação, pode-se utilizar a água armazenada em um reservatório elevado. O armazenamento de água em tanques é muito mais barato do que armazenar energia em baterias (GONZALES, 2010). Depois de cinco anos ou menos, as baterias têm de ser substituídas, enquanto que a

vida de um reservatório de armazenamento bem construído é de várias décadas. O armazenamento em baterias é normalmente justificado quando o rendimento máximo do poço de água durante as horas de sol pleno é insuficiente para satisfazer as necessidades diárias de água e quando é necessário bombear água durante a noite. No longo prazo, pode ser mais barato perfurar outro poço do que adicionar baterias para o armazenamento de energia (KYOCERA, 2002).

Estes sistemas são projetados especialmente para o bombeamento de água de poços, lagoas, rios e açudes e são semelhantes aos sistemas convencionais, com a diferença básica de que o acionamento do motor da bomba é feito por módulos FV (CHILUNDO, 2014).

Figura 3 - Diagrama esquemático de um sistema FV de abastecimento de água.

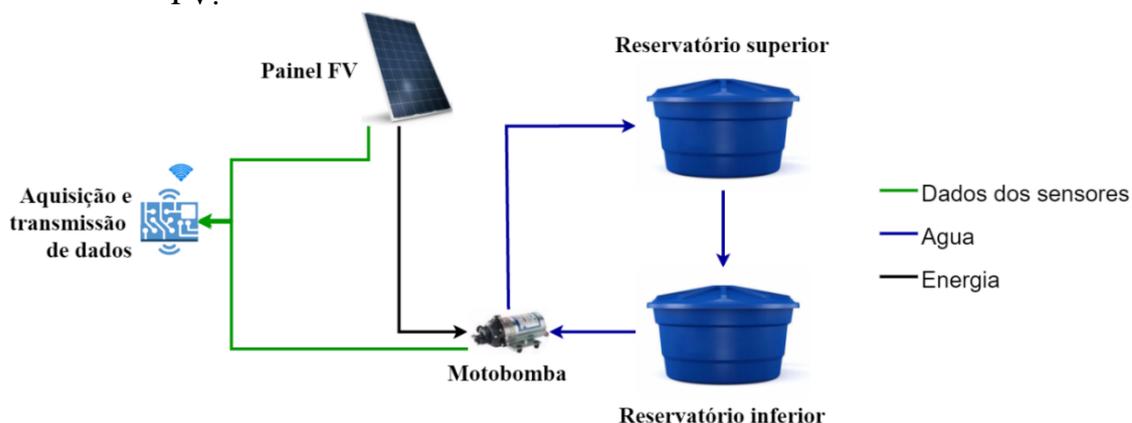


Fonte: Modificado de (CRESESB, 2014).

A Figura 4 mostra a estrutura básica da planta instalada no LEA através do modelo esquemático desenvolvido, composta por quatro subunidades:

1. Suprimento de energia elétrica através de um painel FV.
2. Bombeamento de água mediante o conjunto motobomba;
3. Reservatórios de água superior e inferior;
4. Sistema de aquisição e transmissão de dados e sensores instalados para monitorar o desempenho do sistema.

Figura 4 - Diagrama esquemático da planta de bombeamento FV.



Fonte: Elaborado pelo autor.

### 3.2 Descrição dos materiais e equipamentos

A seguir são explicados os diferentes componentes que fazem parte do sistema proposto.

#### 3.2.1 Módulo fotovoltaico

Para atender à demanda de energia elétrica da planta foi utilizado um módulo FV de células policristalinas marca Yingli Solar modelo YL95P-17b 95 2/3, como ilustrado na Figura 5. Sob condições padrão de teste (Irradiância solar de 1000 W/m<sup>2</sup>, massa de ar 1,5 e temperatura das células de 25°C), o módulo pode atingir uma eficiência de 14,3% com uma tensão de 18,18 V e corrente de 5,23 A na máxima potência. Tabela 2 mostra as especificações técnicas para as condições padrão do painel usado na pesquisa. Para mais informações, ver Apêndice B.

Figura 5 - Painel FV YL95P.



Fonte: Elaborado pelo autor.

Tabela 2 - Especificações técnicas do painel YL95P.

PARÂMETRO	VALOR
Marca	YGE 95
Modelo	YL95P-17b 2/3
Potência máxima (Pmax)	95 W
Eficiência	14,3 %
Tensão de máxima potência	18,18 Vmpp
Corrente de máxima potência	5,23 Impp
Tensão de circuito aberto	22,5 Voc
Corrente de curto circuito	5,59 Isc

Fonte: (YINGLI SOLAR, 2008).

### 3.2.2 Conjunto motobomba e reservatórios de água

A escolha do conjunto motobomba depende da finalidade do sistema e deve ser em fator da máxima eficiência, sob as condições de vazão e altura manométrica utilizadas. Os conjuntos motobomba são acionados por motores de corrente alternada ou corrente contínua (SOUSA, 2014), levando em conta a potência requerida para atingir a altura manométrica desejada, volume de água e capacidade financeira (CHILUNDO, 2014). Em sistemas FV de bombeamento as bombas mais comuns são as centrífugas e volumétricas.

As bombas centrífugas são aquelas que usam a força centrífuga resultante do movimento circular do rotor para deslocamento de água. Em geral, são adequadas para aplicações que exigem grandes volumes de água (elevadas vazões) e pequenas alturas manométricas (reservatórios superficiais ou cisternas), pois para alturas manométricas elevadas esse tipo de bomba apresenta redução na eficiência. Possuem pás ou rotores que giram em alta velocidade, criando pressão e forçando o fluxo de água (CRESESB, 2014) (SOUSA, 2014). As bombas volumétricas têm por característica o comportamento do deslocamento do fluido. São também conhecidas como bombas de deslocamento positivo ou de diafragma, e são robustas, de baixo custo e consumo energético. Em geral, são adequadas quando se deseja atingir grandes alturas manométricas com pequenos ou moderados volumes de água, com a eficiência aumentando com o aumento da altura manométrica. Este tipo de bomba é indicado para operar de forma lenta e eficiente durante todo o dia para vazões na faixa de 0,3 a 40 m<sup>3</sup>/dia e alturas manométricas de 10 a 500 metros. As principais vantagens do uso desta bomba residem na redução dos custos efetivos pela capacidade de acoplamento

direto com o módulo FV para potências de 50 a 400 Wp e capacidade de operar em sobre tensão (SOUSA, 2014)(CHILUNDO, 2014)(CRESESB, 2014).

Para o desenvolvimento da presente pesquisa foi utilizada uma motobomba de deslocamento positivo do tipo diafragma marca SHURflo modelo 8000-443-136, acionada por um motor de ímã permanente de 12 V CC e uma faixa de pressão de 40-60 psi. Figura 6 mostra a motobomba instalada no LEA. Tabela 3 mostra as especificações técnicas do conjunto motobomba utilizada. Para mais informações, ver Apêndice C.

Figura 6 - Conjunto motobomba Shurflo 8000.



Fonte: Elaborado pelo autor.

Tabela 3 - Especificações do desempenho da motobomba SHURflo 8000-443-136.

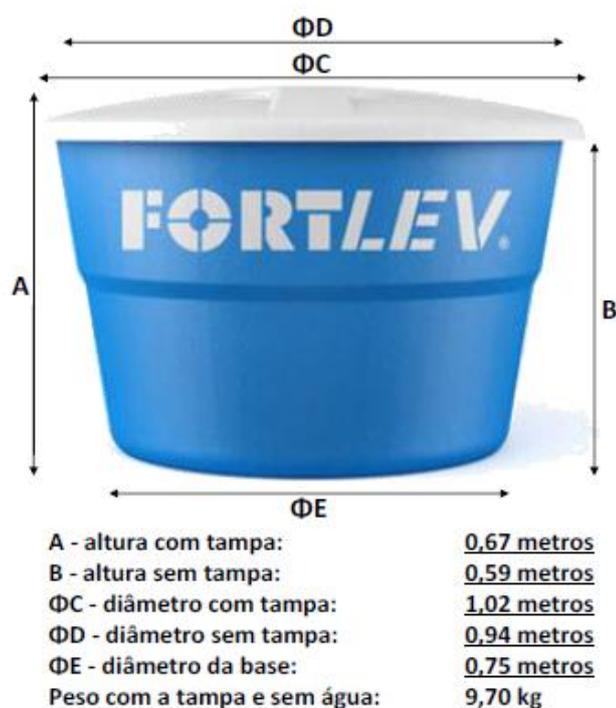
PARÂMETRO	VALOR
Marca	SHURflo
Modelo	8000-443-136
Tensão	12 V DC nominal
Faixa de pressão	40-60 Psi
Temperatura do líquido	170 °F / 77 °C
Peso líquido	4,56 libras (2,07Kg)
Ciclo de trabalho	Contínuo

Fonte: (SHURFLO, 2008).

### 3.2.3 Reservatórios de água

Na presente pesquisa foram usados dois tanques em fibra de vidro de 310 litros de capacidade como reservatórios de água. Conforme manual do fabricante, as caixas utilizadas são fabricadas à base de resina, fibra de vidro e *gelcoat*, garantindo resistência e durabilidade e deixando o produto final imune à proliferação de fungos (MENDONÇA, 2016). As dimensões físicas dos tanques estão apresentadas na Figura 7.

Figura 7 - Dimensões físicas das caixas FORTLEV 310 L.



Fonte: (MENDONÇA, 2016).

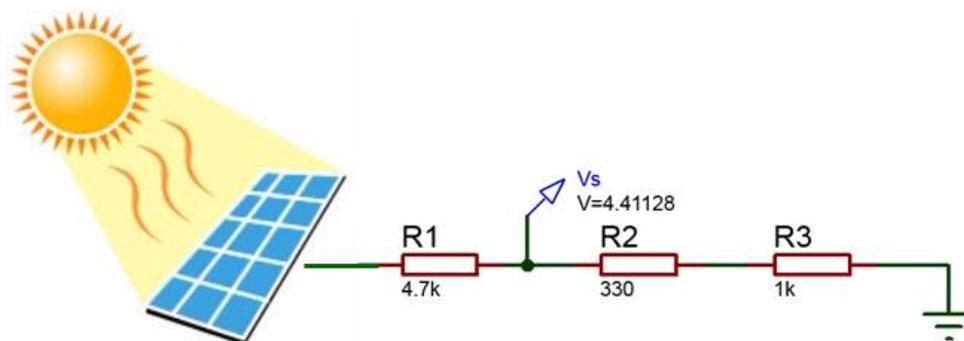
### 3.2.4 Sensores configurados

Os sensores são dispositivos compostos pelo elemento físico sensível e pelo transdutor que traduz as manifestações físicas em sinais elétricos. Nesta pesquisa foram utilizados sensores para medir vazão, pressão, tensão e corrente do painel FV, irradiância, umidade do ar e temperatura ambiente e temperatura do painel.

#### *Tensão e corrente do painel PV*

Como o nível de tensão de saída do módulo pode atingir valores não suportados pelo Conversor A/D da placa SanUSB, a utilização de um sensor de tensão se faz necessária para não danificar a placa de aquisição de dados com uma tensão superior a 5 V. Para essa finalidade, foi utilizado um divisor resistivo como ilustrado na Figura 8. O diagrama do circuito utilizado como divisor é mostrado na Figura 9.

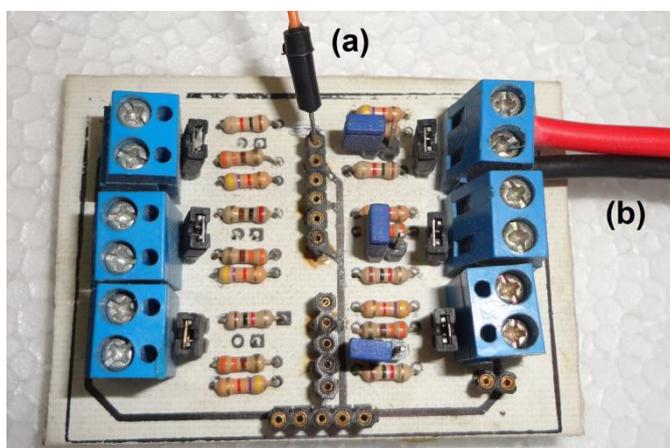
Figura 8 - Divisor de tensão.



Fonte: Elaborado pelo autor.

Figura 9 - Circuito divisor de tensão empregado como sensor de tensão.

a: Saída da tensão para o ADCES. b: Conexão com o Painel FV.



Fonte: Elaborado pelo autor.

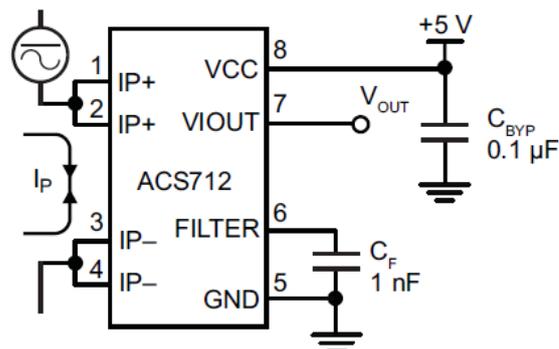
Para a leitura da corrente do módulo, utilizou-se o sensor de efeito Hall com saída em tensão, modelo ACS712, mostrado na Figura 10a. Este sensor está disponível no mercado em 3 modelos diferentes: ACS712ELCTR-05B-T medindo até 5 A, o ACS712ELCTR-20A-T medindo até 20 A e ACS712ELCTR-30A-T medindo até 30A (ALLEGRO MICROSYSTEMS, 2013). Como a sua faixa de leitura de corrente é  $\pm 5$  V, compreende o valor de corrente de curto-circuito do painel. Este sensor trabalha com uma tensão de alimentação de 5 V e tem um erro total de saída de 1,5 % a uma temperatura de 25 °C. O esquemático de ligação do sensor de corrente é mostrado na Figura 10b.

Figura 10 - Sensor de corrente ACS712 e esquemático de ligação do sensor.



Fonte: Elaborado pelo autor.

(a)



Fonte: (ALLEGRO MICROSYSTEMS, 2013)

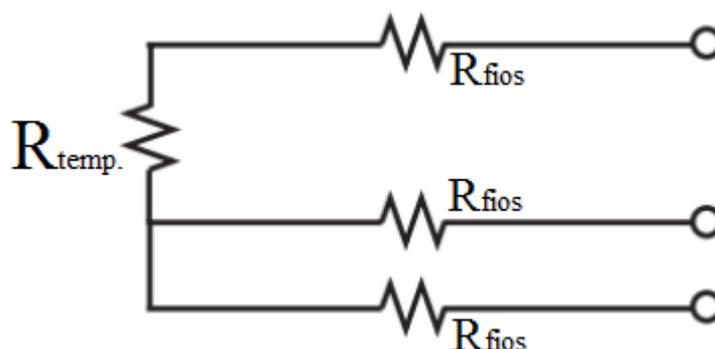
(b)

### ***Temperatura do painel FV***

Com o aumento da temperatura do módulo FV, o desempenho é diminuído, de modo que é necessário realizar uma análise da temperatura do painel. Para realizar esta medição foi utilizado um sensor PT-100 fixado no centro do painel. O sensor PT-100 é um tipo particular de Detector de Temperatura por Resistência (*RTD - Resistance Temperature Detectors*) que mede a temperatura pela correlação da sua resistência elétrica com a temperatura ( $100 \Omega$  a  $0 \text{ }^\circ\text{C}$ ). A maioria destes sensores é construída a partir de uma espiral de fio fino montada num suporte cerâmico ou de vidro. Devido a sua natureza frágil, necessitam ser instalados em bainhas protetoras. Geralmente é considerado como o mais exato sensor de temperatura industrial disponível, além de proporcionar estabilidade aceitável em comparação com outros sensores de temperatura. A relação temperatura x resistência não é linear; porém, com o uso de tabelas, é possível determinar a temperatura correspondente a um determinado valor de resistência. (Ver Apêndice D) (SRCSL, 2016) (ARIAN CONTROL & INSTRUMENTACIÓN, 2016)(ARIAN CONTROL & INSTRUMENTACIÓN, 2016)(ARIAN CONTROL & INSTRUMENTACIÓN, 2016)(GRUPO ECIL, 2016). Cada tipo de sensor de temperatura possui um determinado conjunto de condições para as quais é o mais adequado. PT-100 oferecem várias vantagens: Ampla faixa de temperatura (aproximadamente  $-200$  até  $850 \text{ }^\circ\text{C}$ ), boa precisão e boa permutabilidade (melhor que termopares), e estabilidade em longo prazo. (OMEGA ENGINEERING BRASIL, 2016). Existem 3 modos de conexão para os PT-100: 2 fios, 3 fios e 4 fios. Cada um requer um instrumento leitor diferente. O objetivo é determinar com precisão a resistência elétrica do

sensor  $R(\text{temp.})$  sem influenciar na leitura a resistência de fios  $R(\text{fios})$ . Neste trabalho utilizou-se o modo de conexão de 3 fios, como ilustrado na Figura 11. O sensor foi posicionado na parte de trás do módulo e posteriormente conectado a um transmissor de temperatura da marca NOVUS modelo TxRail, como ilustrado na Figura 12.

Figura 11 - Sensor PT-100 de 3 fios



Fonte: Elaborado pelo autor.

Figura 12 - Sensor PT-100 e Transmissor NOVUS TxRail



Fonte: Elaborado pelo autor.

(a)

Fonte: Elaborado pelo autor.

(b)

A saída do transmissor é corrente de 4-20 mA ou 20-4 mA de tipo 2 fios e linear em relação a temperatura medida pelo sensor selecionado. A resolução da saída é 0,004 mA (12 bits) e a tensão de operação varia numa faixa de 12 a 35 Vcc.

Tabela 4 Tabela 4 apresenta o listado dos sensores aceitos pelo transmissor (NOVUS, 2016).

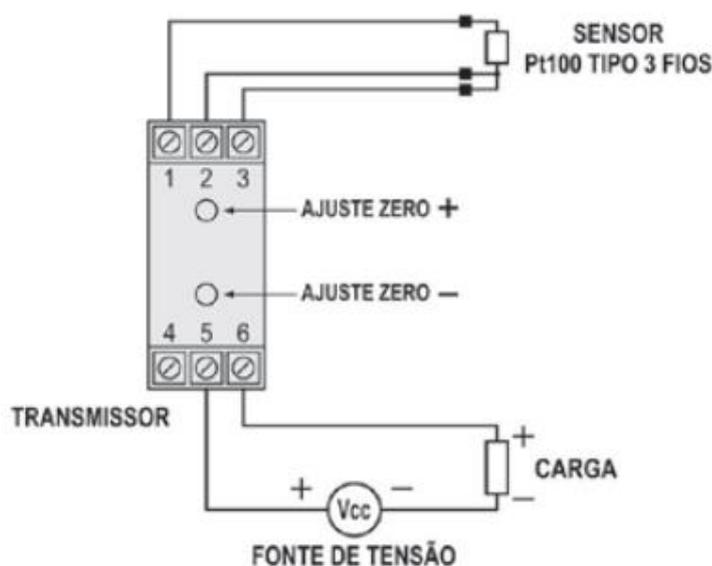
Tabela 4 - Sensores aceitos pelo transmissor

TIPO DE SENSOR	FAIXA MÁXIMA DE MEDIÇÃO	FAIXA MÍNIMA DE MEDIÇÃO DE:
Termopar K	0 a 1370 °C	100 °C
Termopar J	0 a 760 °C	100 °C
Termopar R	0 a 1760 °C	400 °C
Termopar S	0 a 1760 °C	400 °C
Termopar T	0 a 400 °C	100 °C
Termopar N	0 a 1300 °C	100 °C
Termopar E	0 a 720 °C	100 °C
Pt100	-200 a 650 °C	40 °C
Tensão	0 a 50 mV	5 mV

Fonte: (NOVUS, 2016)

Figura 13 mostra as conexões elétricas necessárias para a conexão do sensor de temperatura. Os terminais 1, 2 e 3 são dedicados à conexão de entrada, onde CARGA representa o instrumento medidor de corrente 4-20 mA (indicador, controlador, registrador, etc.). Neste caso foi utilizado o sistema supervisório instalado no laboratório.

Figura 13 - Conexão do sensor de temperatura PT-100.



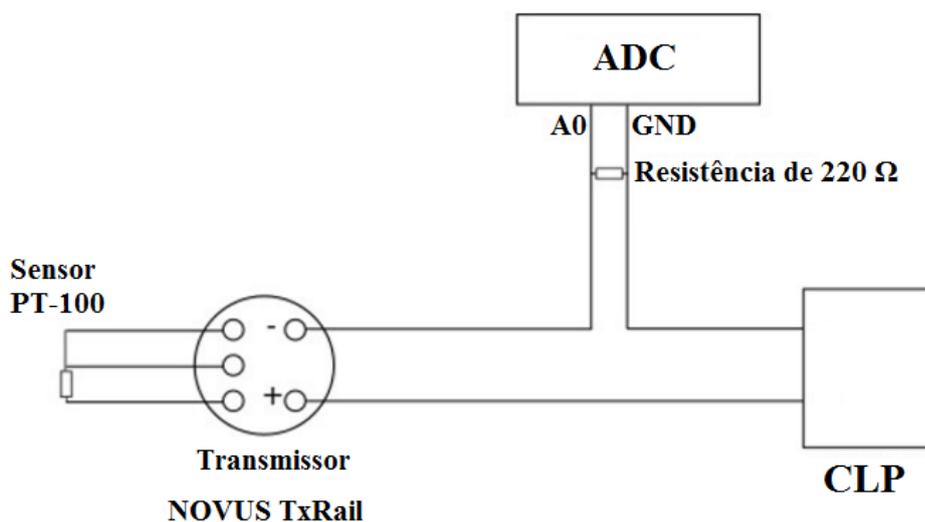
Fonte: (NOVUS, 2016)

A precisão dos dados coletados por um SAD depende basicamente do dispositivo utilizado para a aquisição. A avaliação dessa precisão requer, portanto, um conhecimento dos dados e as fontes que podem contribuir para o erro. O sensor PT-100 apresenta uma resistência ôhmica de 100  $\Omega$  à 0 °C. Além da precisão, outro parâmetro importante é a taxa de

amostragem, que representa a quantidade de vezes por unidade de tempo em que o sinal é medido e convertido. Neste caso, representa o tempo entre duas medições consecutivas de temperatura medida em milissegundos. A medição da temperatura do painel FV foi realizada em paralelo com o sistema de supervisão instalado no laboratório. A leitura de temperatura é realizada a cada um minuto e é armazenado o valor da média das 1000 leituras efetuadas. Desta forma é possível obter valores mais precisos, no entanto este método é mais lento devido ao fato de efetuar várias leituras. No entanto, ele proporciona um maior grau de confiabilidade em relação ao valor armazenado.

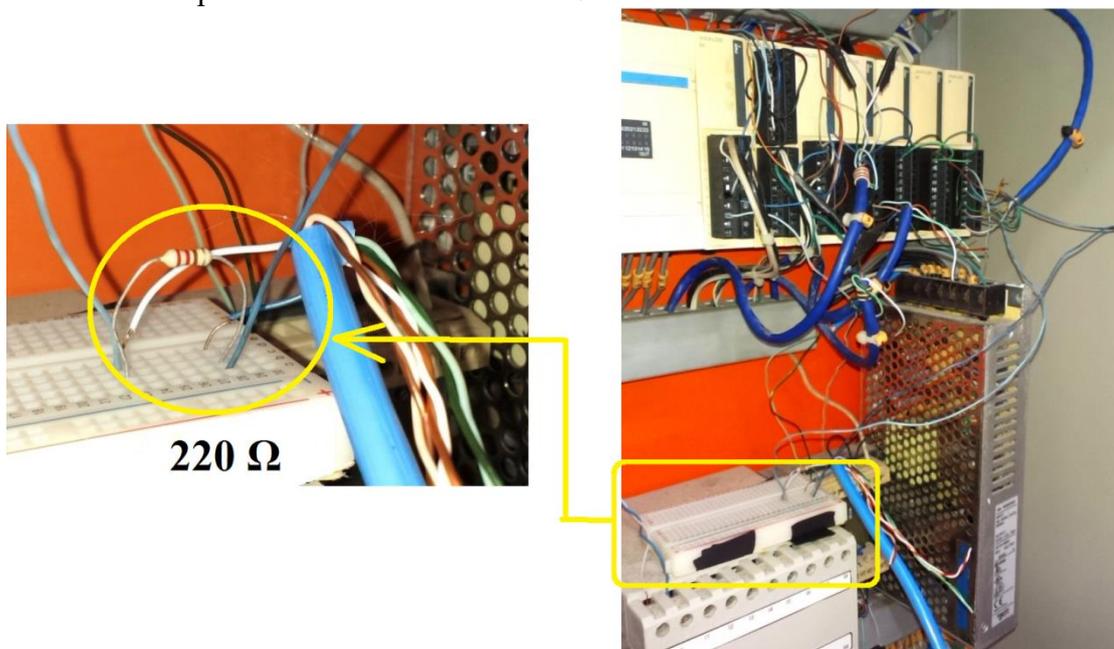
Para converter o sinal de corrente de saída do transmissor num sinal de tensão que o Conversor Analógico/Digital (ADC) possa ler, foi utilizada uma resistência de  $220\ \Omega$ , de modo que, quando o sinal é  $4\ \text{mA}$  será recebido  $0,88\ \text{V}$  no ADC e quando o sinal seja  $20\ \text{mA}$  serão recebidos no ADC  $4,4\ \text{V}$ , ficando no intervalo de medição permitido. A Figura 14 e a Figura 15 ilustram a conexão realizada entre o transmissor e o ADC e o transmissor e o sistema supervisor, respectivamente.

Figura 14 - Conexão entre o transmissor e o ADC



Fonte: Elaborado pelo autor.

Figura 15 - Conexão entre o transmissor e o sistema supervisorio com resistor de 220  $\Omega$ .



Fonte: Elaborado pelo autor.

### *Irradiância*

Para a leitura da irradiância utilizou-se um piranômetro marca Hukseflux modelo LP02 que fornece valores de irradiância global em  $\text{W/m}^2$ , mostrado na Figura 16. O sinal deste sensor é da ordem de microvolts, devendo ser ligado a uma placa amplificadora do sinal, tal como ilustrado na Figura 17 (HUKSEFLUX, 2013). Este sensor tem uma sensibilidade de  $15 \times 10^{-6} \text{ V}/(\text{W/m}^2)$  e uma incerteza de calibração de  $< 1,8 \%$ . O dispositivo fornece um sinal de tensão de saída em função da irradiação presente, limitada a  $2.000 \text{ W/m}^2$ .

Figura 16 - Pirômetro LP02.



Fonte: Elaborado pelo autor.

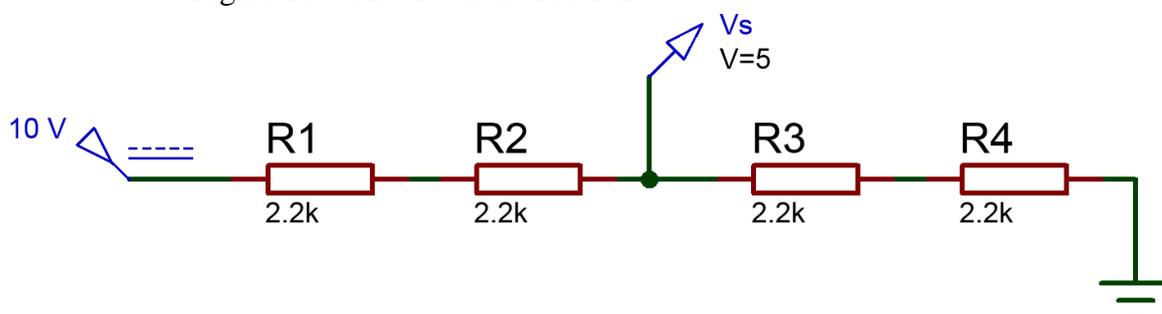
A saída da placa amplificadora varia em uma faixa entre 0 e 10 V dependendo do nível de irradiância, mas o sistema supervisorio lê valores de 0 até 5 V, para tal foi instalado um divisor de tensão com a função de atenuar o valor de tensão para se ajustar ao sistema de aquisição, ilustrado na Figura 18. O pino de saída do divisor de tensão é conectado ao pino de entrada analógica da placa de aquisição de dados.

Figura 17 - Placa amplificadora do sinal.



Fonte: Elaborado pelo autor.

Figura 18 - Circuito divisor resistivo.



Fonte: Elaborado pelo autor.

### *Umidade do ar e temperatura ambiente*

Para as medições de umidade do ar e de temperatura ambiente foi utilizado um sensor digital DHT11 ilustrado na Figura 19a, conectado diretamente ao RaspberryPi. A medição da temperatura com este sensor, varia entre 0 e 50 °C com uma precisão de 2 °C

enquanto a umidade varia dentro de uma gama de 20 a 90% com uma precisão de 5% (D-ROBOTICS UK, 2010). Este sensor trabalha com uma tensão de alimentação entre 3 e 5,5 V. Uma aplicação do sensor é mostrada na Figura 19b, onde MCU representa uma Unidade de microcomputador (*MCU: Micro-computer Unit*). O processo de comunicação e sincronização entre o MCU e o sensor DHT11 é de 4 ms. Os dados enviados consistem de números inteiros e decimais, formando uma sequência de 40 bits com o seguinte formato: 8 bits Dado de umidade (Inteiro) + 8 bits Dado de umidade (Decimal) + 8 bits Dado de temperatura (Inteiro) + 8 bits Dado de umidade (Decimal) + 8 bits Soma de verificação.

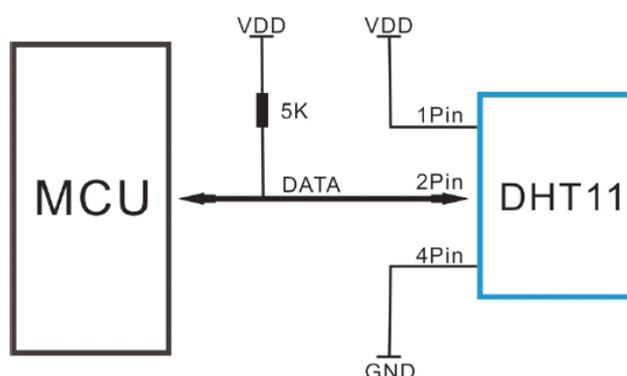
Se a transmissão de dados estiver correta, a soma de verificação deve corresponder aos últimos 8 bits do *string* enviado. O período de amostragem a intervalos não deve ser inferior a 1 segundo.

Figura 19 - Sensor DHT11 e aplicação do sensor.



Fonte: Elaborado pelo autor.

(a)



Fonte: (D-ROBOTICS UK, 2010).

(b)

### ***Pressão e vazão***

Os valores de vazão são medidos pelo sensor de fluxo ultrassônico de marca *IFM ELECTRONIC* modelo SU7000, mostrado na Figura 20. Este sensor proporciona um sinal de saída de 4 a 20 mA ou de 0 a 10 V e tem uma tensão de alimentação que varia entre 19 e 30 VCC. Tem uma faixa de medição de 0 a 50 litros por minuto e uma resolução mínima de 0,1 litros. Suporta uma pressão máxima de 16 bar (253 psi) e uma variação máxima de temperatura do fluido entre -25 e 80 °C. (IFM ELECTRONIC, 2010a).

A pressão na saída da motobomba foi medida com um sensor de marca *IFM ELECTRONIC* modelo PN2024, ilustrado na Figura 21. Este sensor é alimentado por uma tensão de 20 a 30 VCC, suporta uma temperatura entre -25 e 80 °C. Este sensor efetua medições entre -14,5 e 145 psi, gerando um sinal de 4 a 20 mA ou 0 a 10 V como saída. (IFM ELECTRONIC, 2010b).

Figura 20 - Sensor de fluxo volumétrico SU7000.



Fonte: Elaborado pelo autor.

Figura 21 - Sensor de pressão combinada PN2024.



Fonte: Elaborado pelo autor.

### 3.3 Parâmetros calculados

A partir dos dados fornecidos pelos fabricantes, podem ser determinados os parâmetros elétricos e hidráulicos que caracterizam o sistema de bombeamento FV, descritos a seguir:

#### 3.3.1 Parâmetros elétricos

O cálculo da potência fornecida pelo painel FV foi feito de acordo com a expressão:

$$P_{fv} = V \cdot I \quad (1)$$

Onde  $P_{fv}$  representa a Potência em Watts,  $V$  é a tensão e  $I$  a corrente.

No manual do fabricante são fornecidos dados de tensão de circuito aberto ( $V_{oc}$ ), corrente de curto circuito ( $I_{sc}$ ), tensão e corrente de máxima potência ( $V_{mpp}$  e  $I_{mpp}$ , respectivamente) e eficiência ( $\eta_{fv}$ ) do módulo FV. A partir destes parâmetros pode ser calculado o Fator de Forma (FF) definido como a razão entre a máxima potência do módulo e o produto da  $I_{sc}$  com a  $V_{oc}$  (CRESESB, 2014). O FF é uma medida que indica a qualidade das células no módulo FV e expressa o quanto a curva característica I-V se aproxima de um retângulo representativo da máxima potência. O FF varia entre 0,76 e 0,8 em casos de

módulos FV de boa qualidade (MOHANTY; MUNEEER; KOLHE, 2016). O módulo com o FF mais alto (curva I-V mais quadrada) produzirá mais potência. Uma tecnologia de módulo FV ideal produziria uma curva I-V perfeitamente retangular na qual o ponto de potência máxima coincidiria com  $I_{sc}$  e  $V_{oc}$ , para um FF de 1.

O FF é definido pela equação:

$$FF = \frac{V_{mpp} \cdot I_{mpp}}{V_{oc} \cdot I_{sc}} \quad (2)$$

Neste caso, o FF pode ser calculado como a seguir:

$$FF = \frac{18,18 \text{ V} \cdot 5,23 \text{ A}}{22,5 \text{ V} \cdot 5,59 \text{ A}} \quad FF = 0,755964 \quad FF \approx 0,76$$

### 3.3.2 *Parâmetros hidráulicos*

Conforme visto anteriormente, a escolha de um conjunto motobomba leva em consideração como principais fatores, a potência requerida para atingir a altura manométrica necessária, a vazão e a eficiência.

Para o cálculo do volume de água (em litros) bombeado durante o funcionamento do sistema, foi usada a equação:

$$v = Q \cdot t \quad (3)$$

Onde Q representa a vazão em litros por minuto e t o tempo em minutos.

O cálculo da Potência hidráulica depende do volume de água e da altura manométrica total, sendo definida pela equação:

$$P_h = \frac{Q \cdot H \cdot \delta}{4500} \cdot 736 \quad (4)$$

Onde  $P_h$  representa a potência hidráulica em Watts, Q a vazão em litros por minutos, H a altura manométrica em MCA e  $\delta$  o peso específico da água ( $1 \text{ kg/dm}^3$ ).

### 3.3.3 *Eficiências do painel FV, do conjunto motobomba e global*

Outro parâmetro oferecido pelos fabricantes é a eficiência ( $\eta_{fv}$ ) do modulo FV, neste caso o valor fornecido é 14,3 %. A eficiência indica quanto da energia solar incidente no painel é transformada em energia elétrica e está definida pela equação:

$$\eta_{fv} = \frac{P_{fv}}{A \cdot G} = \frac{V \cdot I}{A \cdot G} \cdot 100 \% \quad (5)$$

Onde  $A$  representa a área do módulo em  $m^2$ ,  $G$  a irradiância solar incidente em  $W/m^2$ ,  $V$  a tensão e  $I$  a corrente. Neste caso a área do módulo é  $0,6666 m^2$ .

A eficiência da motobomba foi calculada pela equação:

$$\eta_{mb} = \frac{P_h}{P_m} \cdot 100 \% \quad (6)$$

Onde  $P_h$  representa a potência hidráulica e  $P_m$  a potência elétrica do motor, ambas em Watts.

A eficiência global do sistema FV de bombeamento foi obtida através do produto entre a eficiência do módulo FV e a eficiência da motobomba, conforme segue na equação:

$$\eta_g = \frac{\eta_{fv} \cdot \eta_{mb}}{100} \quad (7)$$

Onde  $\eta_g$  representa a Eficiência global do sistema,  $\eta_{mb}$  a Eficiência da motobomba e  $\eta_{fv}$  a Eficiência fotovoltaica, todas em percentagem (%).

### 3.4 Conclusões parciais

Este capítulo apresentou a descrição da planta de bombeamento fotovoltaico usada para desenvolver a presente pesquisa. Mostrou-se uma descrição dos diferentes sensores configurados bem como os parâmetros necessários para atender o desempenho do sistema.

## 4 SISTEMA EMBARCADO PARA MONITORAMENTO ONLINE PROPOSTO

Neste capítulo é descrito o sistema de monitoramento *online* proposto na pesquisa bem como os princípios de funcionamento. São descritas:

- A arquitetura geral do sistema junto com as diferentes etapas da pesquisa;
- O funcionamento do sistema desde a coleta dos dados dos sensores até o armazenamento das medições no servidor *online*;
- A conexão entre a placa SanUSB e o Raspberry Pi (Rpi);
- As características dos serviços em nuvem bem como a descrição da interface gráfica para a visualização dos dados armazenados.

### 4.1 Arquitetura geral do sistema

O sistema proposto é baseado na arquitetura “*Plataforma como Serviço*”, do inglês *PaaS - Platform as a Service*, como ilustrado na Figura 22. Esta arquitetura é uma das camadas de serviços pertencente ao modelo de monitoramento em nuvem (*Cloud Computing*). A camada superior do modelo é “*Software como Serviço*”, do inglês *SaaS - Software as a Service*, que corresponde a um modelo de distribuição de aplicações no qual o fornecedor recebe pedidos dos clientes e os torna disponíveis através da Web. A camada interior é chamada “*Infraestrutura como Serviço*”, do inglês *IaaS – Infrastructure as a Service*. IaaS representa um modelo de autosserviço para o acesso, monitoramento e gerenciamento de infraestruturas de centros de dados, onde os usuários podem comprar recursos informáticos com base no consumo. Nesta camada é fornecido tanto o *hardware* quanto o *software*, ambos virtuais. O cliente tem acesso aos componentes virtualizados a fim de construir as suas próprias plataformas de TI. A definição inclui a oferta de espaço virtual de servidores, conexões de rede, largura de faixa, endereços IP e balanceadores de carga (VMWARE, 2011) (KEPES, 2013).

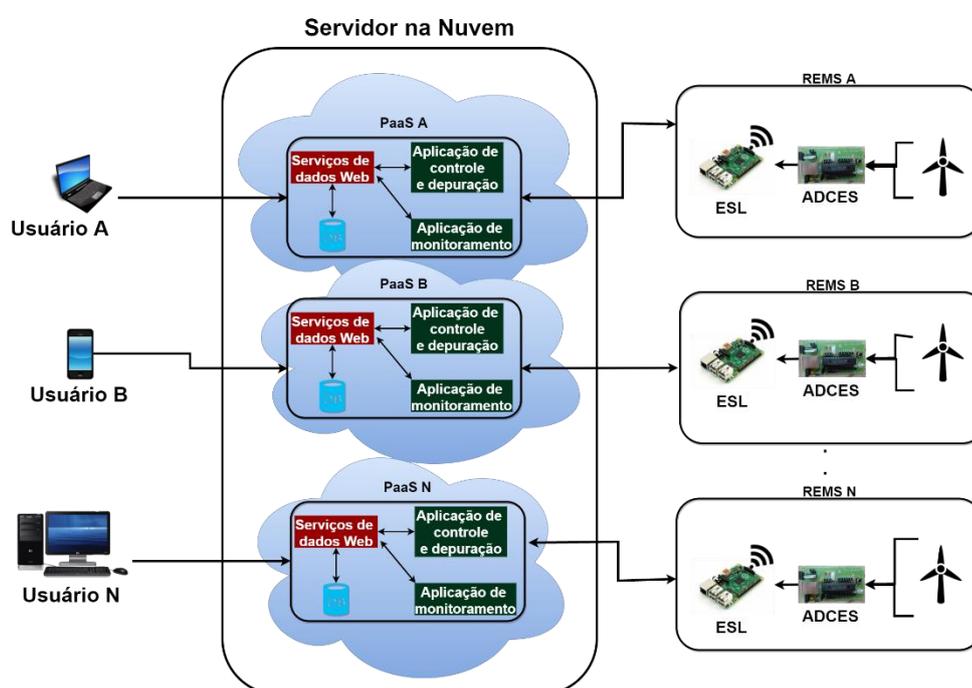
O Instituto Nacional de Padrões e Tecnologia (*NIST - National Institute of Standards and Technology*) define o monitoramento em nuvem como um modelo para permitir o acesso sob demanda a um conjunto compartilhado de recursos de computação configuráveis (por exemplo, redes, servidores, armazenamento, aplicações e serviços) que podem ser fornecidos e liberados com esforço de gerenciamento mínimo ou pouca interação do provedor de serviço (MELL; GRANCE, 2011).

Figura 22 - Camadas do modelo de monitoramento em nuvem



Fonte: Adaptado de VMWARE (VMWARE, 2011)

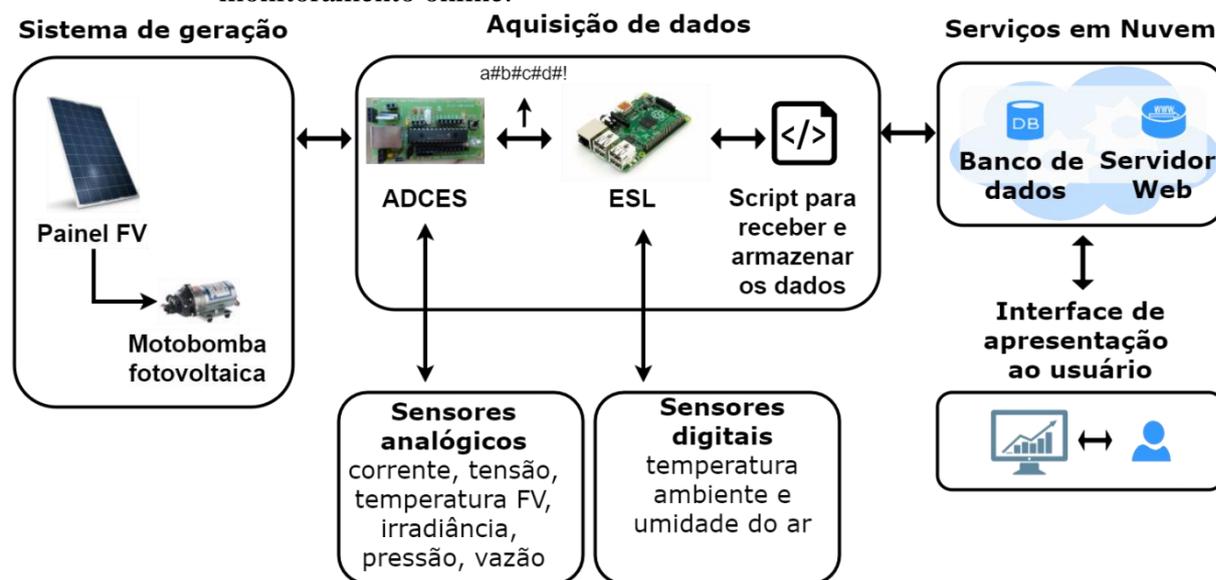
Por outro lado, *PaaS* é definido como um conjunto de serviços centrados na tecnologia e apresentados para o desenvolvimento e implementação de aplicações oferecidas diretamente para os desenvolvedores através de um catálogo de serviços na Internet (VMWARE, 2011). O provedor fornece ferramentas de *hardware* e *software* para seus usuários como um serviço, libertando-os de ter que instalar e configurar *hardware* e *software* próprios para desenvolver ou executar uma nova aplicação. A Figura 23 mostra o esquema *PaaS* aplicado ao monitoramento *online* proposto de fontes de energias renováveis (*REMS-Renewable Energy Monitoring System*).

Figura 23 - Ilustração do *PaaS* e do *REMS* proposto.

Fonte: Elaborado pelo autor.

O sistema proposto permite a criação de diversos perfis de PaaS em um banco de dados na nuvem. O perfil PaaS é gerado pelo servidor online (por exemplo, PaaS A, B ou N), após o cadastro de uma senha pelo usuário. Permite desenvolver, compilar, depurar e/ou testar uma aplicação na nuvem. Dessa forma, um usuário qualquer (A, B ou N) pode criar e configurar na nuvem e em tempo real um perfil de monitoramento, bem como realizar a configuração do firmware e das características de funcionamento do sistema embarcado de monitoramento online. A arquitetura geral do sistema proposto é apresentada na Figura 24.

Figura 24 - Arquitetura do sistema embarcado proposto para monitoramento online.



Fonte: Elaborado pelo autor.

A arquitetura desenvolvida é constituída por um sistema de geração, um bloco de aquisição de dados, sensores, serviços em nuvem e a apresentação ao usuário. No presente estudo é utilizada uma planta de bombeamento alimentada por painel FV. O bloco de aquisição de dados é composto por um sistema embarcado baseado em microcontrolador PIC18F2550, um sistema embarcado Linux (*Embedded Linux System - ELS*) desenvolvido na placa Rpi que através do acesso à Internet, que pode ser *WiFi* ou *Ethernet*, enviam os dados coletados dos sensores analógicos e digitais configurados no sistema a uma interface entre a aquisição de dados e os serviços em nuvem. O *software* embarcado desenvolvido é gratuito e pode ser obtido após o registro de um novo perfil *PaaS* no servidor em Nuvem. O sistema é programado para funcionar de forma descentralizada e distribuída, permitindo que múltiplos usuários e múltiplos *REMS* se comuniquem ao mesmo tempo e usando o mesmo servidor na

nuvem. No caso de configuração ou de modificação do *REMS*, são necessários conhecimentos básicos em instalação de aplicações no sistema operacional *Linux*.

Neste estudo foi feita a criação e configuração de um perfil de usuário para realizar diferentes configurações *online* dos componentes do sistema, como por exemplo, a configuração da rede sem fio do *Rpi* ou a gravação *online* do *ADCES*. A interface de comunicação foi desenvolvida para receber e tratar os dados enviados pelo *Rpi* antes do seu armazenamento no banco de dados. Os serviços em nuvem são compostos pelo servidor Web e o banco de dados *online*. Estes elementos são responsáveis pelo armazenamento das medições feitas a cada minuto e pela interação entre usuários/administradores e a planta monitorada. O servidor Web é um espaço na nuvem que fornece o serviço de hospedagem das páginas do site desenvolvido bem como o tratamento das informações registradas no banco de dados. Estas informações podem ser consultadas por qualquer dispositivo conectado à Internet desde que o usuário tenha cadastrado usuário e senha previamente no site.

O *software* desenvolvido para o sistema de monitoramento e os *softwares* de apoio utilizados como banco de dados e linguagem de programação são *softwares* livres sem restrições de cópia ou uso. Foi desenvolvida também uma interface amigável para uma melhor interação usuário/sistema de aquisição de dados. O usuário pode acessar os gráficos dos diferentes sensores configurados no sistema bem como os logs armazenados para cada um dos sensores.

## 4.2 Aquisição de dados

O processo de aquisição de dados está dividido nas seguintes etapas:

- Aquisição de dados através dos sensores analógicos e digitais: tensão e corrente do painel FV, temperatura na parte traseira do painel, temperatura ambiente, irradiância, umidade do ar, pressão e vazão;
- Envio dos dados pelo *ADCES* ao *Rpi* e condicionamento dos dados recebidos;
- Transmissão dos dados para um servidor em nuvem com o intuito de gerar gráficos e manipulá-los de acordo com as necessidades do usuário.

Os dados dos sensores analógicos recebidos pelo *ADCES* são tratados e enviados de forma serial em uma sequência de caracteres pré-definida. Primeiramente, este *string* é recebida pelo *Rpi* junto com a informação dos sensores digitais, em seguida é tratada para extrair e concatenar os valores dos diferentes sensores e, por fim são enviados a uma página Web que atua como uma interface de comunicação entre o *Rpi* e o banco de dados.

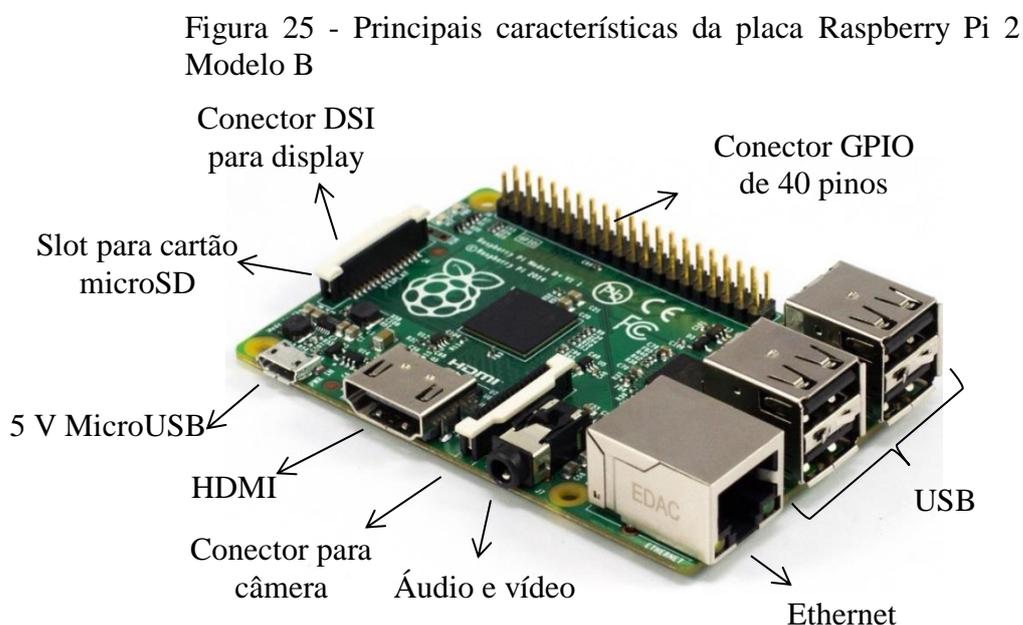
#### 4.2.1 Sistema Embarcado Conversor Analógico/Digital (ADCES)

O ADCES é composto pelo sistema de desenvolvimento SanUSB. Esta ferramenta computacional é composta por um gerenciador gravado no microcontrolador e uma interface gráfica utilizada no computador para gravar o novo *firmware* na memória de programa flash do microcontrolador via USB (PEREIRA, 2014). SanUSB é constituído por *software* e *hardware* básico da família PIC18Fxx5x com interface USB. Como não há necessidade de remover o microcontrolador da placa para a atualização do *firmware*, a ferramenta mostra-se eficiente no desenvolvimento de projetos reais. O *software* de gravação de microcontroladores USB é multiplataforma, ou seja, executável no *Windows*®, *Mac OSX* e no *Linux*, e é reconhecido automaticamente pelos sistemas operacionais sem a necessidade de instalar nenhum driver JUCÁ *et al.*, (2009). Desta forma, esta ferramenta possibilita que a compilação, a gravação e a simulação real de um programa, possam ser feitos de forma rápida e eficaz a partir da conexão do circuito microcontrolado a um sistema computacional via USB. Considerando-se as vantagens da ferramenta, foi usada a interface gráfica e o *software* de gravação para transferência do *firmware* diretamente do ELS para o PIC do ADCES utilizando o protocolo HID (*Human Interface Device*) de comunicação através da porta USB. O ADCES proposto é também um sistema embarcado RISC (*Reduced Instruction Set Computer*) e permite atualização *online* do *firmware* via nuvem. Desta forma, é aplicável a projetos de Internet das Coisas (*IoT - Internet of Things*), com acesso remoto via *Internet* e não somente *Intranet*.

#### 4.2.2 Plataforma Raspberry Pi

A plataforma *Raspberry* foi desenvolvida pela Fundação *Raspberry Pi* com o objetivo de melhorar a aprendizagem na área computacional de crianças e adultos. É um computador pequeno de baixo custo, baseado em código livre e com interfaces para vários periféricos. O modelo usado neste trabalho é o *Raspberry Pi 2 Modelo B* que foi projetado para substituir a versão anterior, o *Raspberry Pi Modelo B+*. Em comparação com a versão anterior, este modelo possui um processador *ARM Cortex-A7* com quatro núcleos de 900 MHz. Outras características dele são a memória *RAM* de 1 GB compartilhada com a *GPU* (*Graphics Processing Unit*), núcleo gráfico VideoCore IV 3D, 40 pinos entrada/saída (*I/O*) de uso geral (*GPIO - General-purpose input/output*), 4 portas com interface USB 2.0 para conectar o teclado e o mouse utilizado em computadores de mesa, porta *HDMI* (*High Definition Multimedia Interface*), porta *Ethernet* com interface RJ-45, interface para cartão

micro *SD* (*Secure Digital*), interface para câmera e monitor bem como a alimentação de 5 V e saída de áudio (RASPBerry FOUNDATION, 2016). A Figura 25 apresenta o modelo usado neste projeto junto com as principais características citadas.



Fonte: Elaborado pelo autor.

### *Instalação e configuração do sistema operativo*

O sistema operacional (SO) oficial suportado pela Fundação *Raspberry Pi* é o *Raspbian*. Este SO oferece várias ferramentas pré-instaladas para a educação, a programação e o uso geral (RASPBerry FOUNDATION, 2016). Alguns exemplos de aplicações pré-instaladas são *Python*, *Scratch*, *Sonic Pi*, *Java*, *Mathematica*, entre outras. Outro aspecto interessante do projeto *Raspbian* são as variantes não oficiais desenvolvidas pelos usuários da comunidade de código aberto. Como exemplos de SO desenvolvidos suportados pela plataforma *Rpi*, podem ser citados: *Arch Linux ARM* (ARCH LINUX ARM, 2016), *OpenELEC* (OPENELEC, 2016), *Pidora* (PIDORA, 2016), *OpenWrt para Raspberry Pi* (OPENWRT, 2016), *Raspberry Digital Signage* (RASPBerry DIGITAL SIGNAGE, 2016), *RISC OS Pi* (RISC OS, 2016), entre outros. Cada um deles foi desenvolvido com finalidades e características específicas, mas com o objetivo comum de ser usado na plataforma *Rpi*.

Na presente pesquisa foi usada a distribuição *Raspbian* baseada em *Debian Jessie* com data de lançamento Fevereiro de 2016. Para instalar o SO é necessário baixar a imagem do site oficial da Fundação. Esta imagem pode ser baixada como arquivo compactado com extensão *.zip*, posteriormente é necessário descompactar o arquivo para obter uma imagem

com extensão .img. Esse arquivo tem que ser gravado na micro *SD* usando uma ferramenta de gravação de imagens. As instruções mudam dependendo do SO onde vai ser usada a ferramenta. Depois de ser gravada a imagem do SO na micro *SD*, é necessário procurar a direção IP obtida pelo *Rpi* após ter sido ligado ao cabo da rede local. O sistema é iniciado e está pronto para ser usado com as configurações padrão tais como usuário pi e senha raspberry. Por razões de segurança, é aconselhável mudar essa senha para uma mais segura.

#### *Instalação e configuração dos principais pacotes e bibliotecas para a comunicação com os outros componentes do sistema*

Para o correto desenvolvimento deste trabalho, foram utilizados um grupo de bibliotecas e pacotes para garantir a comunicação entre todos os componentes do sistema. Para acessar o *Rpi* foram testados os protocolos *SSH (Secure Shell)* e *VNC (Virtual Network Computing)* para garantir o acesso por console e em ambiente gráfico, respectivamente.

*SSH* é um protocolo que facilita a comunicação segura entre dois sistemas usando uma arquitetura cliente/servidor e permite que os usuários se conectem a um host remoto. Ao contrário de outros protocolos de comunicação remota como *FTP (File Transfer Protocol)* ou *Telnet*, *SSH* criptografa a sessão de login, tornando impossível para intrusos a coleta de senhas não criptografadas (RED HAT ENTERPRISE LINUX 4, 2016a).

Por outro lado, o protocolo *VNC* é usado para gerenciamento remoto de sistemas usando ambientes gráficos, já que garante a interação completa com o computador conectado. Este é um protocolo que permite a visualização de interfaces gráficas remotas através de uma conexão segura. O usuário que acessa ao host remoto pode ver e acessar todo o conteúdo de outro computador. Este protocolo é amplamente usado por profissionais que oferecem serviços de assistência técnica a outros usuários (CANALTECH, 2016).

Para garantir a interconexão entre computadores com sistemas operacionais *Microsoft Windows* e *Linux* e compartilhar arquivos e pastas, foi instalado e configurado o pacote Samba (SAMBA, 2016). *Samba* é uma implementação *open source* do protocolo *SMB (Server Message Block)* que pode agir como um controlador de domínio primário (*Primary Domain Controller - PDC*) e pode atuar como uma cópia de segurança do controlador de domínio (*Backup Domain Controller - BDC*) para um *PDC* baseado no *Samba* (RED HAT ENTERPRISE LINUX 4, 2016b).

Para converter os programas de linguagem de alto nível escritos em linguagem de máquina foi usado o compilador *GCC (GNU Compiler Collection)* (GCC, 2016). *GCC* é uma

distribuição integrada de compiladores para várias das principais linguagens de programação. Linguagens como o *C*, *C++*, *C#* e *Java* são exemplos destas linguagens. Neste trabalho foi usada a linguagem *C* para desenvolver o programa do microcontrolador e os programas para coletar os dados dos sensores analógicos e digitais no *Rpi*. *GCC* foi originalmente escrito como o compilador para o sistema operacional *GNU (GNU's Not Unix)* (GNU, 2016). O projeto *GNU* foi lançado por Richard Stallman em 1983, como um sistema operacional que seria montado por pessoas trabalhando em conjunto para alcançar a liberdade de todos os usuários de *softwares*.

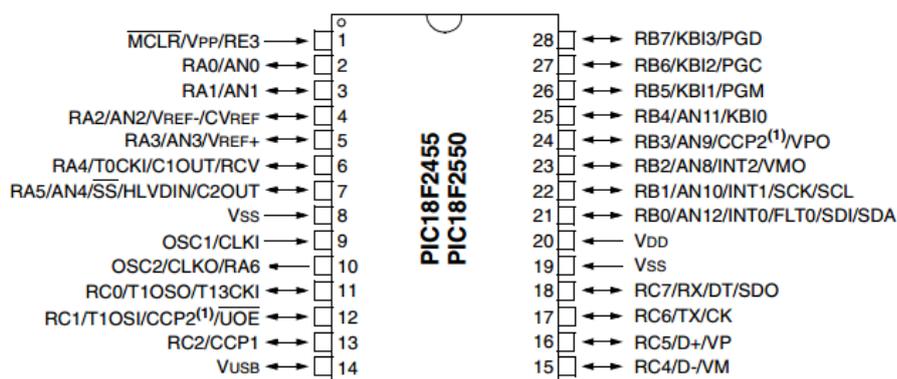
*WiringPi* é uma biblioteca escrita em *C* para ser utilizada no *Rpi* que inclui um utilitário de linha de comando *GPIO* que pode ser usado para programar e configurar os pinos *GPIO* (WIRINGPI, 2016). Neste trabalho foi usada a biblioteca *WiringPi* para acessar e trabalhar com as portas *GPIO* do *Rpi*. Também foi usado *CURL* (CURL, 2016) como ferramenta de linha de comando *open source*, biblioteca para a construção da sintaxe *URL* e para a transferência a nuvem dos dados coletados pelos sensores.

#### *Coleta e tratamento de dados para enviar à interface web como string pré-formatado*

Nesta pesquisa foi utilizado o microcontrolador PIC18F2550 produzido por Microchip Technology Inc. O Controlador Integrado de Periféricos (PIC - *Peripheral Interface Controller*) é um circuito integrado que contém todos os circuitos necessários para realizar um completo sistema digital programável em um único dispositivo. Internamente dispõe de todos os dispositivos típicos de um sistema microprocessado: CPU, memórias ROM e RAM, linhas de I/O para controlar dispositivos externos ou receber pulsos de sensores e chaves, dispositivos auxiliares ao funcionamento como gerador de clock, bus, contador, entre outros.

A Figura 26 apresenta a disposição física dos pinos do PIC18F2550. Por se tratar de um microcontrolador multifunções, um único pino pode ser configurado para executar diferentes tarefas. As principais características do microcontrolador são descritas na Tabela 5.

Figura 26 - Distribuição física dos pinos do PIC18F2550



Fonte: (MICROCHIP, 2006).

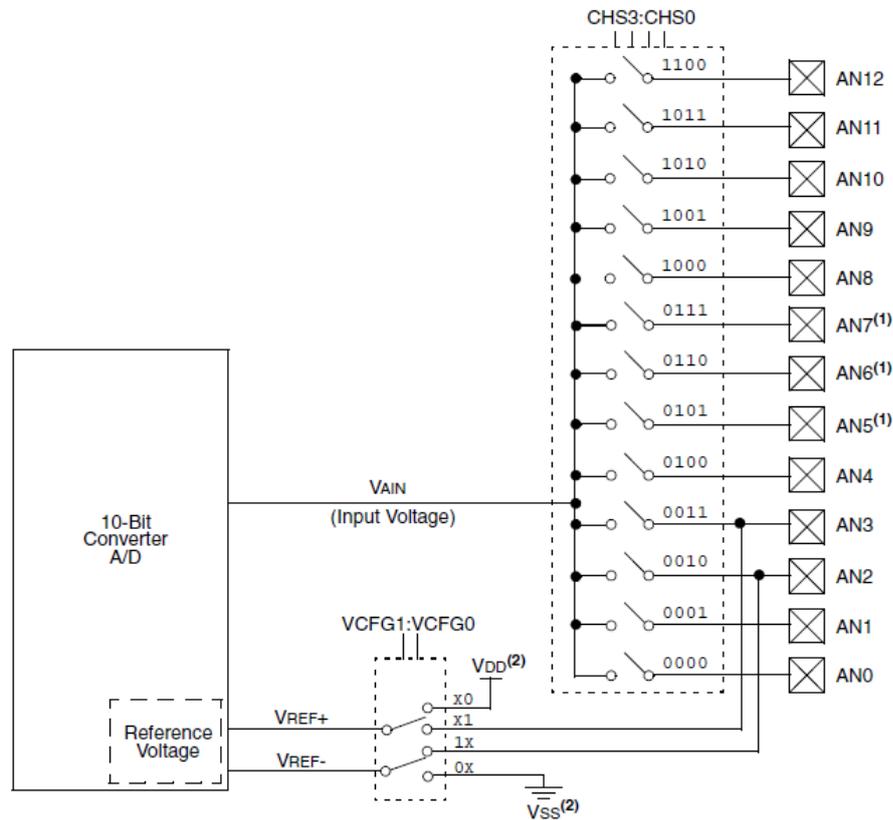
Tabela 5 - Características do PIC18F2550

Parâmetro	PIC18F2550
Tensão de alimentação	5 V
Corrente máxima de entrada	250 mA
Corrente máxima de saída nos pinos	200 mA
Memoria FLASH	32 kBytes
Memoria SRAM	2048 kBytes
Memoria EEPROM	256 Bytes

Fonte: (MICROCHIP, 2006).

O módulo Conversor Analógico/Digital (ADC) do PIC18F2550 tem 10 entradas e é capaz de medir a tensão analógica em cada canal e converter a tensão medida em um valor digital de 10 bits. Este módulo incorpora um tempo de aquisição programável, permitindo que um canal seja selecionado e uma conversão seja iniciada, sem esperar por um período de amostragem e, portanto, reduzindo a sobrecarga de código (MICROCHIP, 2006). A Figura 27 apresenta o diagrama em blocos do ADC.

Figura 27 - Diagrama de blocos do ADC

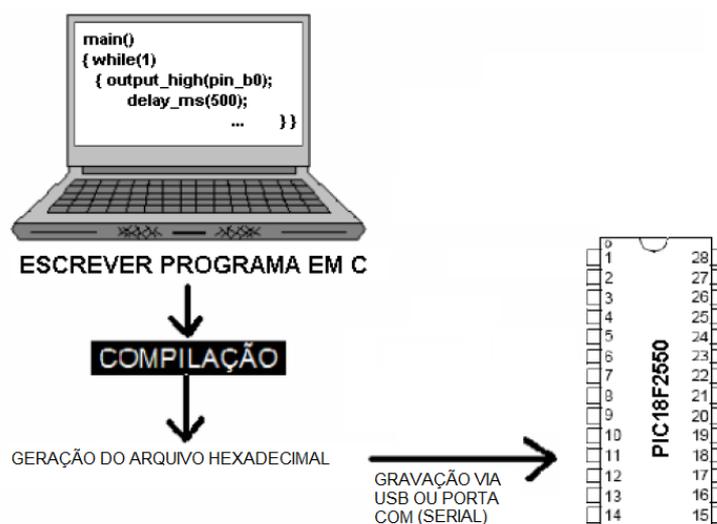


- Note 1:** Channels AN5 through AN7 are not available on 28-pin devices.  
**Note 2:** I/O pins have diode protection to VDD and VSS.

Fonte: (MICROCHIP, 2006).

Para programar o microcontrolador são necessários conhecimentos básicos sobre programação. As linguagens mais usadas são: Assembly, C, Visual Basic e Pascal. Como ambiente de desenvolvimentos desta pesquisa foi usado o ambiente de desenvolvimento (IDE) *MPLABX* da *Microchip* e o compilador C18. Este é constituído de um IDE gráfico que pode ser executado em qualquer plataforma Windows. Este ambiente permitirá uma programação dos PICs em linguagem C. Neste software são escritas as instruções lógicas, para depois compilar o programa e gerar o arquivo no formato *\*.hex* (hexadecimal). Após ter gerado o arquivo, o próximo passo é gravá-lo no microcontrolador. A Figura 28 apresenta um esquemático do processo de gravação do PIC18F2550.

Figura 28 - Processo de gravação do microcontrolador

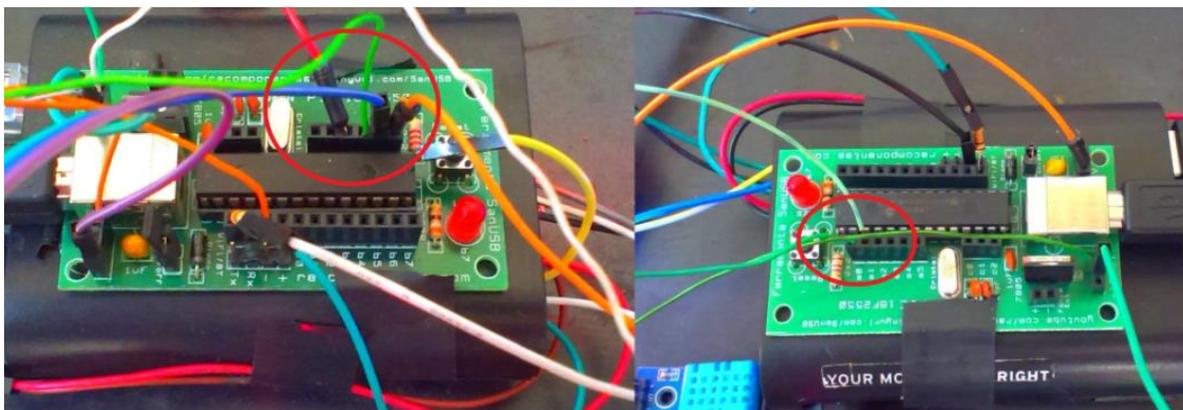


Fonte: Elaborado pelo autor.

Os sensores analógicos estão conectados diretamente à placa SanUSB seguindo a distribuição ilustrada na Figura 29.

Figura 29 - Conexão dos sensores analógicos à placa SanUSB.

(a) Tensão, corrente e irradiância. (b) Pressão, vazão e temperatura do painel.



(a)

(b)

Fonte: Elaborado pelo autor.

A Tabela 6 apresenta a distribuição das entradas do PIC e os sensores que são conectadas nelas.

Tabela 6 – Conexão dos sensores nas entradas do PIC.

Sensor	Entradas do PIC1	Sensor	Entradas do PIC2
Tensão	AN0	Pressão	AN0
Corrente	AN1	Vazão	AN1
Irradiância	AN2	Temperatura do painel FV	AN2

Fonte: Elaborado pelo autor.

Para coletar os dados destes sensores, foi gravado no microcontrolador um programa para medir os sinais das entradas analógicas via serial e envia-as via USB ao Rpi. Primeiramente, em nosso programa usamos o comando de leitura analógica *le\_AD10bits(PIN)* da biblioteca *SanUSB1.h* para fazer a leitura do valor das entradas analógicas, onde PIN representa o número do canal analógico do PIC a ser lido. A seguir é mostrada uma parte do código usado para medir tensão, corrente e irradiância.

```
result0 = le_AD10bits(0); tempo_ms(50); // Tensão
result1 = le_AD10bits(1); tempo_ms(50); // Corrente
result2 = le_AD10bits(2); tempo_ms(50); // Irradiância
```

O programa lê qual é o valor do sinal nos canais analógicos, que pode ser um valor de tensão entre 0 V e 5 V e converte o valor lido em um número entre 0 e 1023. Finalmente, usando o comando *swputc(' letra')* e *FloattoAscSerial(tensao, str)* são enviados os valores convertidos de cada sensor medido. Como o nível de tensão de saída do modulo FV e os valores de pressão e vazão do conjunto motobomba podem atingir valores não suportados pelo ADC do PIC18F2550, foi utilizado um divisor resistivo para não danificar a placa por sobretensão. No programa gravado no microcontrolador, foi realizado um tratamento dos valores lidos sendo multiplicados pelo valor do divisor resistivo, para obter os valores reais medidos. Em relação a temperatura do painel, foram efetuadas 1000 leituras e feita uma media com o objetivo de obter valores mais precisos. Os dados são enviados via comunicação serial ao Rpi em um *string* com o seguinte formato: **a# b# e# f# g# h# !**, onde o símbolo # representa a posição da medição de cada um dos sensores medidos e o símbolo ! representa o final do *string*. A seguir é mostrada uma parte do código usado para enviar os valores medidos de tensão, corrente e irradiância.

```
swputc('a');
FloattoAscSerial(tensao, str); // Valor real de tensão
swputc('b');
```

```

FloattoAscSerial(corrente, str);    // Valor real de corrente
swputc('c');
FloattoAscSerial(irradiancia, str); // Valor real de irradiância
swputc('!');                        //Indica fim da medição

```

O sensor digital DHT11 para medir temperatura ambiente e umidade do ar, é conectado diretamente ao Rpi como ilustrado na Figura 30.

Figura 30 - Conexão do sensor digital DHT11 à placa Rpi.



Fonte: Elaborado pelo autor.

Para coletar os valores de umidade do ar e temperatura ambiente, foi desenvolvido outro programa em linguagem C, desta vez para ser executado no *Rpi*. Este programa permite gravar a data da medição, junto com o valor das variáveis de umidade e temperatura ambiente e armazenar esta informação em um arquivo *dht11.log* para o posterior tratamento deles.

Finalmente foi desenvolvido um programa principal em linguagem C para receber pela serial o *string* com os dados dos sensores analógicos enviados pelo PIC e ler o arquivo *dht11.log* para obter os valores de umidade e temperatura ambiente. Este programa recebe o *string*, faz um ciclo para extrair e concatenar cada um dos bytes recebidos e armazená-los em uma variável para cada sensor. O ciclo é executado até que o símbolo “!” seja recebido, o que significa o final da linha de medições. Depois de serem recebidos todos os bytes, é construída uma cadeia *URL* que fazendo uso da biblioteca *CURL*, é enviada à interface *PHP* antes de ser armazenada no banco de dados.

#### *Automatização das tarefas planificadas com ferramenta cron*

A administração e suporte de serviços requer conhecimento de equipamentos, aplicativos e usuários, para manter a par dos avanços tecnológicos, revisões e correções de

programas instalados e as necessidades em geral da entidade. Para executar essas tarefas, a equipe de *TI* precisa automatizar a maioria dos procedimentos repetitivos para evitar erros tipográficos ou conceituais e melhorar a gestão global de aplicações (LABRADOR, 2010). No *Linux*, as tarefas podem ser configuradas para se executar automaticamente dentro de um determinado período de tempo, em uma data específica ou quando a média de carga do sistema estiver abaixo de um determinado número (RED HAT ENTERPRISE LINUX 5, 2016). Alguns exemplos de tarefas que devem ser automatizadas são:

- Verificação de espaço livre em discos.
- Gestão de contas de usuário e revisão periódica das cotas de disco.
- Criação, verificação e restauração de *bakups* de segurança.
- Verificar e registrar o desempenho geral do sistema e da rede de computadores.
- Criação de alertas de segurança como, por exemplo, verificação de mudanças no sistema ou detecção de intrusão (LABRADOR, 2010).

Na automatização de tarefas de administração, geralmente é usada a programação de *scripts* para ajudar à equipe de *TI* a executar tarefas de forma mais eficiente. Esses *scripts* podem ser escritos e executados usando linguagens interpretadas pelo *shell* mais conhecidos como intérprete de comandos do sistema. Estes *scripts* permitem juntar comandos do sistema com estruturas de controle de fluxo para oferecer um ambiente mais vasto de oportunidades de administração de tarefas agendada (ESTEVE; BOLDRITO, 2004). Outro esquema comum é o uso de ferramentas de compilação e depuração de linguagens de alto nível como, por exemplo C.

Neste trabalho foi usada a automatização de tarefas usando o Cron como uma ferramenta básica de gerenciamento e planificação de trabalhos para o administrador do sistema de monitoramento proposto. Os programas ou scripts envolvidos na seção anterior foram compilados e planificados com o uso da ferramenta Cron para ser executados a cada minuto. Cron é uma ferramenta que pode ser usada para programar a execução de trabalhos recorrentes de acordo com uma combinação do tempo (horário), dia do mês, o mês, dia da semana e o comando a executar (RED HAT ENTERPRISE LINUX 5, 2016). A configuração principal desta ferramenta é baseada na edição do arquivo `/etc/crontab` e a adição da linha correspondente ao script a planificar com o formato mostrado na Tabela 7.

Tabela 7 - Formato de configuração do crontab.

<b>Mínuto</b>	<b>Hora</b>	<b>Dia</b>	<b>Mês</b>	<b>Dia da semana</b>	<b>Comando</b>
Número inteiro de 0 a 59	Número inteiro de 0 a 23	Número inteiro de 1 a 31 (deve ser um dia válido se o mês for especificado)	Número inteiro de 1 a 12 (ou o nome abreviado do mês, como jan ou fev)	Número inteiro de 0 a 7, onde 0 ou 7 representa o Domingo (ou a abreviação do dia da semana, como sun ou mon)	O comando a executar (Pode ser um comando do sistema ou um script personalizado)

Fonte: Elaborado pelo autor.

No caso que seja necessário especificar todos os valores possíveis, pode ser usado um asterisco (\*) para qualquer um dos valores anteriores. Para especificar uma lista, pode ser usada uma lista de valores separados por vírgulas (.). Um hífen (-) entre os inteiros especifica uma escala de números inteiros.

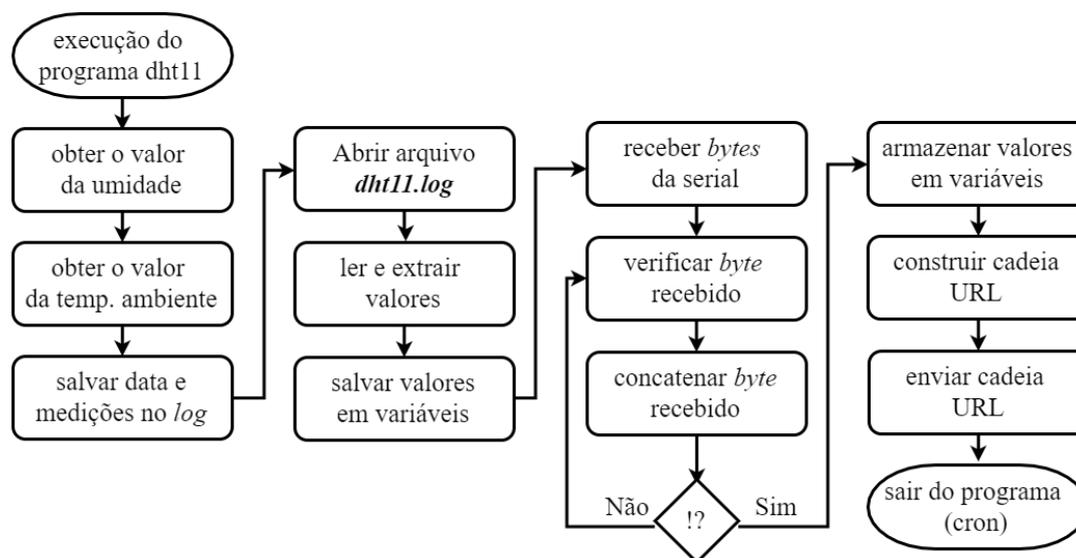
Neste trabalho, foi usada a configuração mostrada na Tabela 8. A Figura 31 explica o processo de comunicação entre os *scripts* planificados a ser executados com a ferramenta Cron.

Tabela 8 - Configuração do *crontab* usada na pesquisa.

<b>Mínuto</b>	<b>Hora</b>	<b>Dia</b>	<b>Mês</b>	<b>Dia da semana</b>	<b>Comando</b>
*/*	*	*	*	*	echo \$(date +\% Y\%m\%d\%H\%M\%S),`/home/share/miDHT11` >> /home/share/dht11.log echo `/home/share/miPrograma` >> /home/share/programa.log

Fonte: Elaborado pelo autor.

Figura 31 - Fluxograma de tratamento dos dados dos sensores.



Fonte: Elaborado pelo autor.

#### 4.2.3 Descrição da conexão entre a placa SanUSB e o Raspberry Pi

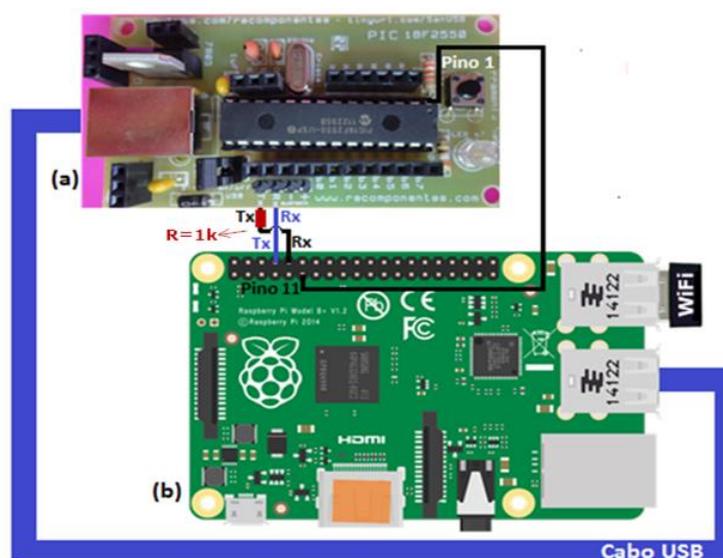
Neste trabalho foram utilizados como meio de comunicação a comunicação serial e a interface USB como formas de conexão entre a placa SanUSB e o *Rpi* e a biblioteca *WiringPi* para acessar os pinos *GPIO* do *Rpi*. Para realizar a comunicação serial entre um microcontrolador *PIC* e o *ELS* baseado em *Raspberry Pi*, é preciso utilizar os pinos *Ground*, o *GPIO* 14 (TX) e o *GPIO* 15 (RX) como ilustrado na Figura 32.

Figura 32 - Distribuição dos pinos GPIO do Raspberry Pi 2 Modelo B.

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Nesta pesquisa foi usado o próprio *Rpi* para realizar a gravação do microcontrolador da placa PIC SanUSB, cujos pinos de conexão para gravação via USB e comunicação serial são ilustrados na Figura 33. A transferência de programas é normalmente efetuada através de um hardware de gravação específico e via cabo. Através desta ferramenta, é possível efetuar a descarga de programas para o microcontrolador, conectado à USB do *Rpi*, de forma online, através do sistema embarcado *Rpi* conectado à Internet.

Figura 33 - Conexão serial entre a placa SanUSB (a) e o Raspberry Pi (b).



Fonte: Elaborado pelo autor.

### 4.3 Envio e armazenamento em nuvem

O objetivo da presente pesquisa é desenvolver um sistema embarcado *Linux* para o monitoramento em Nuvem de uma planta de bombeamento de água alimentada por painel FV. Para cumprir com este objetivo, foi desenvolvido um ambiente em *software* livre baseado na arquitetura *PaaS* para o monitoramento da planta. Esta arquitetura permite a administração própria de aplicações e dados dentro da infraestrutura informática oferecida nesta tecnologia.

Basicamente neste tipo de arquitetura, são oferecidos recursos em nuvem para o desenvolvimento e gerenciamento de aplicações e armazenamento de dados; enquanto a instalação do sistema operacional, ambientes de virtualização e as camadas físicas de *hardware* é responsabilidade do fornecedor do serviço. A Tabela 9 mostra os serviços por tipo de arquitetura que correspondem à gestão própria, indicado com fonte **negrita**, e os serviços que são da responsabilidade do provedor, marcados em fonte *itálica*.

Tabela 9 - Serviços por tipo de arquitetura do modelo de monitoramento em nuvem.

Tipo de gestão	IaaS	PaaS	SaaS
Aplicações	Própria	Própria	Provedor
Dados			
Runtime		Provedor	
Middleware			
Sistema Operacional			
Virtualização	Provedor		
Servidores			
Armazenamento			
Rede			

Fonte: Elaborado pelo autor.

Para aproveitar as vantagens oferecidas pela *PaaS*, o processo de monitoramento foi dividido nas seguintes etapas:

1. Aquisição de dados através dos sensores: tensão e corrente FV, temperatura do painel e ambiente, irradiância e umidade do ar, pressão e vazão da planta de bombeamento.
2. Envio dos dados pelo *ADCES* ao *Rpi* e condicionamento dos dados recebidos.
3. Transmissão dos dados para um servidor em Nuvem para geração de gráficos e manipulação dos dados de acordo com a necessidade do usuário.

A transmissão dos dados foi feita fazendo uso dos serviços em Nuvem classificados da seguinte forma:

#### 4.3.1 Servidor Web

O servidor web é composto pelos módulos: Apache e PHP que são fornecidos pela arquitetura *PaaS*.

##### *Apache*

O Apache é um servidor *HTTP* (*Hypertext Transfer Protocol*) de código fonte aberto e de licenciamento livre que funciona em *Linux*, sistemas operacionais *Unix*-derivados, *Windows*, *Novell Netware* e outras plataformas (ALCANCE LIBRE, 2016). O Apache é um projeto desenvolvido para criar um servidor Web configurável, robusto e de alto desempenho com código fonte disponível livremente na Internet. Além disso, centenas de usuários contribuíram ideias, código e documentação para ajudar a melhorar o projeto (APACHE, 2016).

Algumas características que fazem que este programa seja o preferido entre os administradores de sistemas são:

1. Possui suporte a *scripts CGI (Common Gateway Interface)* usando linguagens como *Perl, PHP, Shell Script, ASP*, etc.
2. Suporte a autorização de acesso, podendo ser especificadas restrições de acesso separadamente para cada endereço/arquivo/diretório acessado no servidor.
3. Autenticação requerendo um nome de usuário e senha válidos para acesso a alguma página/subdiretório/arquivo, suportando criptografia.
4. Negociação de conteúdo, permitindo a exibição da página Web no idioma requisitado pelo navegador.
5. Suporte a tipos *MIME (Multipurpose Internet Mail Extensions)*, personalização de *logs* e mensagens de erro.
6. Suporte a virtual *hosting* (É possível servir 2 ou mais páginas com endereços diferentes (virtual IP) ou portas ou nomes virtuais através do mesmo processo ou usar mais de um processo para controlar mais de um endereço).
7. Suporte a servidor *Proxy, FTP e HTTP* com limite de acesso, flexivelmente configuráveis.
8. Suporte redirecionamentos baseados em URLs para endereços internos (SILVA, 2010).

## *PHP*

O *PHP (PHP: Hypertext Preprocessor)* é uma linguagem de programação interpretada que pode ser usada junto com código *HTML* para o desenvolvimento de páginas web. O objetivo principal da linguagem é permitir a desenvolvedores escreverem páginas que serão geradas dinamicamente e rapidamente com uma sintaxe da linguagem similar a *C, Java e Perl* (PHP, 2016a).

Algumas características desta linguagem de programação são as seguintes:

1. O código é executado no servidor, gerando *HTML* e enviado ao cliente para receber o resultado da execução do script.
2. Extrema simplicidade para o iniciante, mas oferece muitos recursos avançados para programadores profissionais.
3. Pode ser usado em todos os principais sistemas operacionais, incluindo Linux, muitas variantes de *Unix* (incluindo *HP-UX, Solaris e OpenBSD*), *Microsoft*

*Windows, Mac OS X, RISC OS. PHP* suporta a maioria dos servidores web incluindo *Apache, IIS (Internet Information Services)* e outros.

4. Criação de imagens, arquivos *PDF* e até mesmo filmes em flash gerados durante a execução. Pode-se facilmente gerar qualquer tipo de texto, como *XHTML* e *XML*.
5. Amplo suporte para diferentes bancos de dados.
6. Tem suporte para comunicar-se com outros serviços usando protocolos como o *LDAP (Lightweight Directory Access Protocol)*, *IMAP (Internet Message Access Protocol)*, *SNMP (Simple Network Management Protocol)*, *NNTP (Network New Transfer Protocol)*, *POP3 (Post Office Protocol, version 3)*, *HTTP*, *COM (Communication port in Windows)* e outros (PHP, 2016b).

#### **4.3.2 Banco de dados**

Para o armazenamento e processamento da informação em nuvem foi usado *MySQL* como *SGBD (Sistema de Gestão de Banco de Dados)*. *MySQL* é um banco de dados relacional que armazena dados em tabelas em vez de uma única locação o que permite maior velocidade e flexibilidade. Utiliza a Linguagem de Consulta Estruturada (*SQL - Structured Query Language*) como interface. O *software MySQL* é de Licença Dupla onde os usuários podem optar por usar o *software* como um produto *open source* sob os termos da *GNU General Public License* ou podem comprar uma licença comercial padrão da *Oracle* (MYSQL, 2016).

Algumas características deste *software* são:

1. O objetivo principal é a robustez e a velocidade de processamento.
2. Suporta elevado número de tipos de dados para as colunas.
3. Oferece portabilidade entre sistemas, permitindo trabalhar em diferentes plataformas e sistemas operacionais.
4. Aproveita as vantagens dos sistemas de multiprocessadores graças à sua implementação *multithreading*.
5. Apresenta um sistema flexível de senhas e gerenciamento de usuários com nível de segurança de dados.
6. Suporta mensagens de erro em diferentes idiomas TOLEDO ALMA *et al.*, (2016).

### 4.3.3 Gravação em nuvem dos componentes do sistema

Os dados dos sensores são tratados e enviados em uma sequência de caracteres pré-definida. Depois de ser construída a cadeia com os valores de todos os sensores, a informação é postada e armazenada no banco de dados num servidor em Nuvem através do Rpi para sua apresentação aos usuários. Esta cadeia é enviada para uma página *PHP* no servidor Web que atua como uma interface de comunicação entre o *Rpi* e o banco de dados, como pode ser visto na Figura 34.

Figura 34 - Interface de comunicação para gravação no banco de dados.

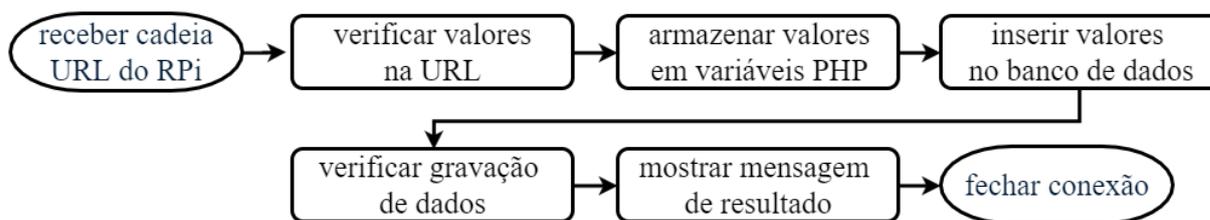


Fonte: Elaborado pelo autor.

A figura mostra uma camada prévia onde são definidos os parâmetros para a comunicação com o banco de dados como o nome do servidor onde está hospedado o banco de dados, o nome de usuário e a senha para efetuar a conexão e o banco de dados específico.

A configuração através de uma interface de comunicação oferece maiores vantagens sobre uma configuração de envio e inserção diretamente sobre o banco de dados, pois a última configuração não permite uma comunicação com servidores gratuitos na Internet. Por outro lado, com a configuração adotada, os dados são enviados para o servidor Web e este é responsável pelo armazenamento no banco de dados configurado. Isto permite trabalhar com um sistema modular, ou seja, podem ser feitas alterações nos diferentes blocos do sistema sem afetar a configuração dos outros componentes. A Figura 35 mostra o fluxograma do programa que recebe os dados.

Figura 35 - Fluxograma do programa para receber dados.



Fonte: Elaborado pelo autor.

Os dados coletados pelo sistema estão publicados em um servidor na Internet para serem visualizados remotamente através de uma página armazenada neste servidor. Para fins de teste, está sendo utilizado um servidor gratuito para envio e recepção dos dados. Embora este tipo de serviço tenha uma série de limitações tais como a implementação de linguagens de script e a utilização de banco de dados, este pode ser utilizado durante desenvolvimento e testes.

#### 4.4 Conclusões parciais

Neste capítulo foram apresentadas as principais características e vantagens do sistema de monitoramento proposto, sendo estas o baixo custo e emprego de *hardware* e *software* abertos. O uso da placa *Rpi* simplificou o processamento e proporcionou um circuito em tamanho portátil utilizando o *Linux* como sistema operacional e C linguagem de programação. O desenvolvimento de programas para as diversas funcionalidades demonstrou a possibilidade de construção de uma solução eficaz e de baixo custo, onde o usuário pode acrescentar funcionalidades e configurações de acordo com suas necessidades. O monitoramento remoto via Internet elimina a necessidade da instalação de qualquer *software* no computador que acessará o servidor.

## 5 APLICAÇÃO E RESULTADOS

Neste capítulo é apresentado o princípio de funcionamento do sistema proposto e a descrição da interface Web desenvolvida para a interação entre o usuário e o SAD. Finalmente, são mostrados os resultados obtidos experimentalmente com uso da arquitetura implementada.

### 5.1 Princípio de operação do sistema

A planta de microgeração descentralizada utilizada na presente pesquisa está instalada no LEA-UFC e consiste em um sistema de bombeamento de água acionado por um painel FV. São coletados e monitorados dados de tensão e corrente FV, irradiância, temperatura ambiente e do painel FV e pressão e vazão do conjunto motobomba. O módulo fornece potência máxima de 95 Wp, tensão de circuito aberto de 22,5 V e corrente de curto-circuito de 5,59 A. Com referência ao conjunto motobomba, para uma tensão de alimentação de 12 V e pressão de 10 psi, pode ser alcançada uma vazão de 6,3 litros por minuto e uma corrente de 3,4 A. O *software* livre de monitoramento *online* e de aquisição sem fio de dados desenvolvido possibilita acompanhar o desempenho da planta de bombeamento FV em tempo real, tanto no laboratório quanto pela Internet.

Como visto anteriormente, a arquitetura do sistema é dividida em dois subsistemas: um deles inserido no equipamento para a aquisição dos dados e o outro no servidor em Nuvem. A informação é enviada fazendo uso da biblioteca *CURL* à interface Web “*getParameterFromRPi.php*” que é responsável de salvar os dados recebidos. Esse script está armazenado no servidor Web e contém a configuração dos parâmetros para a conexão com o banco de dados. Com o sucesso da inserção dos dados, a página PHP gera uma mensagem de “Dados inseridos corretamente”, mas se ocorrer algum erro será exibido uma mensagem de erro através do comando *mysql\_error()*. Este comando retorna o texto do erro da última função do MySQL.

Para postar valores no servidor *online*, é enviada desde o Rpi uma instrução no seguinte formato:

```
"http://HOST/ftpmonitor/getParameterFromRPi.php?action=send1&voltage=18.02&current=1.87&temperature2=31.0&humidity=32.0&temperature1=61.08&radiation=1004.78&pressure=3.94&flowrate=6.52!&date=2016-08-30%2009:33:02"
```

Onde *HOST* representa o nome do servidor onde estão sendo enviados os dados coletados e cada um dos respectivos sensores envia os seus valores instantâneos.

## 5.2 Interface gráfica

Para a visualização dos dados foi desenvolvida uma aplicação Web num servidor na nuvem com a linguagem de programação *PHP* em conjunto com uma base de dados *MySQL*. Para o acesso ao sistema é necessário apenas um navegador e digitar a *URL* <http://sanusb.org/ftpmonitor/>. Imediatamente é mostrada a página de autenticação do sistema para digitar usuário e senha. Após o usuário ter passado pela etapa de autenticação, será redirecionado para a página inicial do sistema. O menu localizado na parte superior da página é composto por 4 itens principais além das opções para mudar o idioma (português, espanhol e inglês), alterar a senha do usuário conectado e sair do sistema, conforme a Figura 36.

Figura 36 - Menu do sistema.

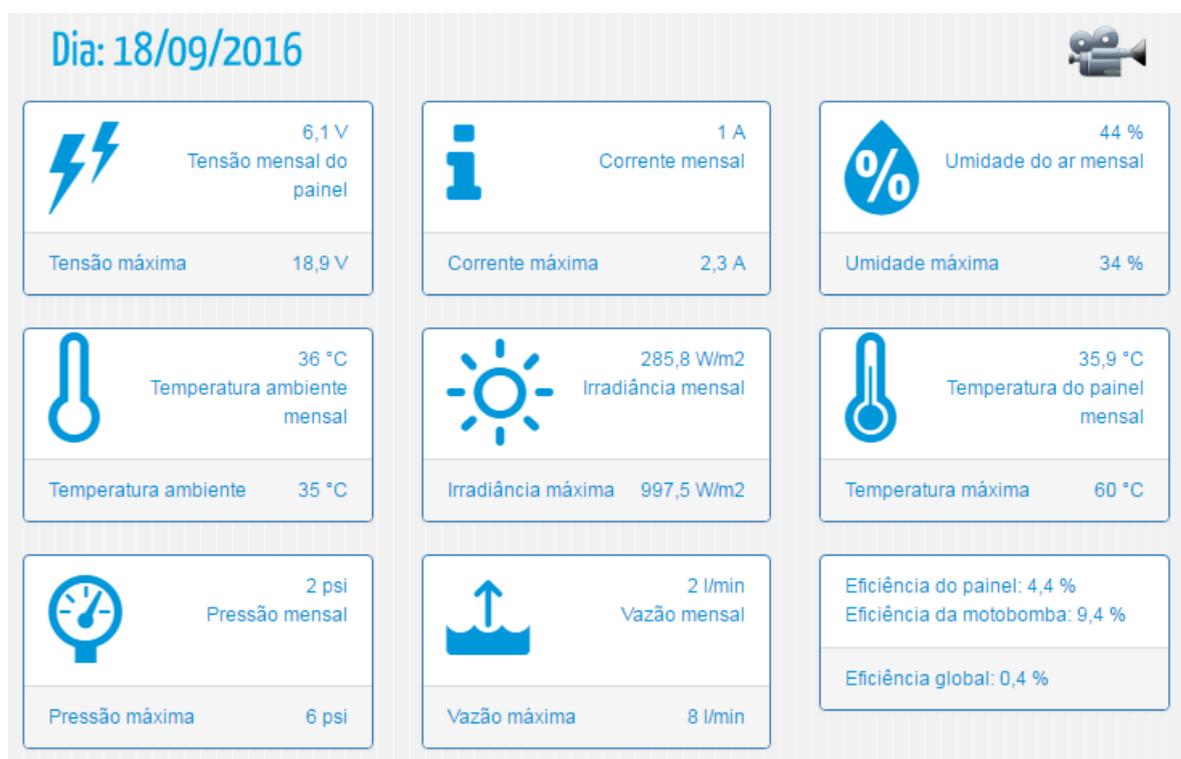


Fonte: Elaborado pelo autor.

Na guia “Monitoramento” pode ser consultado o gráfico de cada sensor. É possível também acessar os gráficos em tempo real e os logs de monitoramento de todos os sensores configurados, bem como exportar e imprimir as informações para diferentes formatos (*PNG*, *JPEG*, *PDF* e *SVG* para gráficos; *PDF* e *XLS* para *logs*), permitindo filtrar por períodos de data. A opção “Gráficos gerais” exibe representações das médias diárias e os maiores e menores valores registrados para cada sensor, além de gráficos para conhecer o comportamento diário das variáveis técnicas envolvidas. A opção “Administração” oferece a lista dos usuários e sensores configurados no sistema e facilita a gestão dos diferentes indicadores do banco de dados, bem como funções de manutenção e suporte necessárias para o funcionamento adequado do sistema. Os novos usuários recebem uma notificação do sistema após a sua criação. O programa de monitoramento envia um e-mail com os dados do novo usuário junto com a senha inicial definida pelo administrador. No primeiro acesso ao sistema, a alteração da senha é obrigatória verificando assim a segurança e a integridade do sistema. Na guia “Equipe de pesquisa” pode ser consultado o Currículo Lattes de cada um dos participantes da pesquisa.

A Figura 37 ilustra o painel principal do sistema onde é mostrado um resumo com os valores máximos a cada um minuto e uma média mensal para cada sensor.

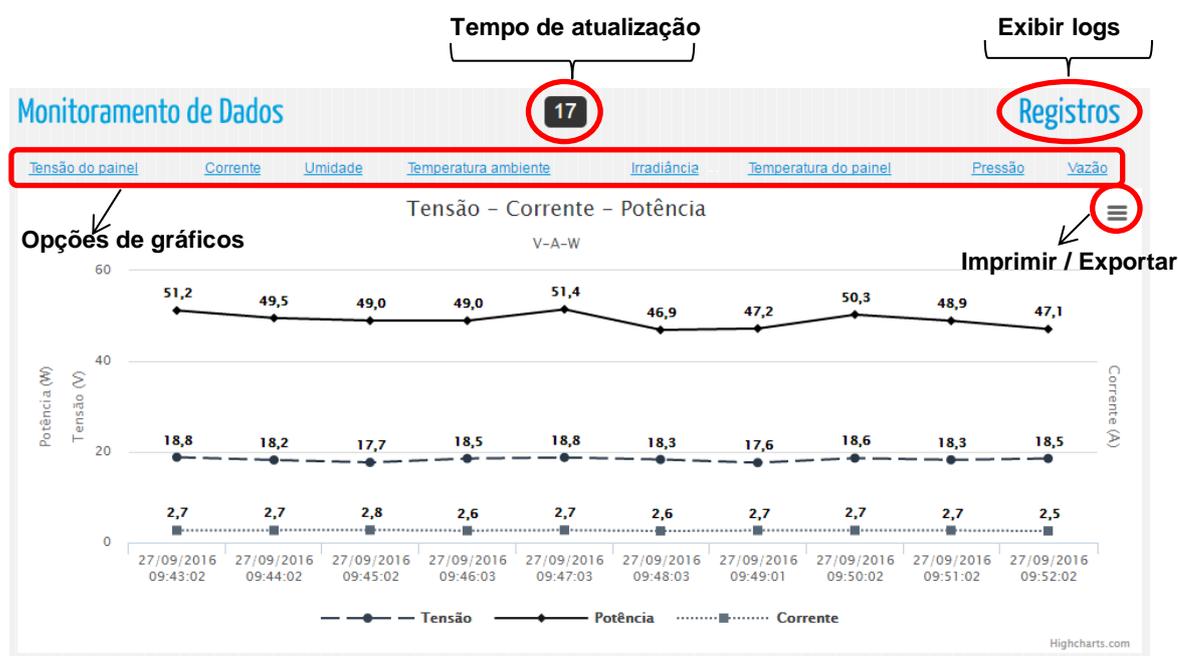
Figura 37 - Painel principal do sistema



Fonte: Elaborado pelo autor.

Ao escolher uma opção, é gerado um gráfico que apresenta os valores instantâneos do sensor selecionado que estão armazenados no banco de dados, conforme Figura 38. Além da informação desse sensor, são mostrados também os dados de outro sensor. Por exemplo, no caso em que é escolhida a opção “Tensão”, é exibida também a informação relacionada com a corrente gerada pelo painel bem como a potência FV calculada. A tela mostrada proporciona um gráfico com as medições a cada minuto do sensor escolhido. Também podem ser acessados os logs com todas as medições coletadas. Através do sistema de monitoramento é possível consultar os dados armazenados no banco de dados *online* através de smartphones ou PCs. Foi usada *Highcharts* como biblioteca feita em *Java Script* para a geração dos gráficos, que são reiniciados a cada 30 segundos automaticamente, mas se houver necessidade de reiniciar o gráfico antes, pode clicar na opção do navegador.

Figura 38 - Exemplo de gráfico gerado pelo SAD



Fonte: Elaborado pelo autor.

Podem ser consultados os dados armazenados de um dia específico selecionando uma data válida. O gráfico será atualizado exibindo um valor representativo da média horária calculada para os valores dos sensores escolhidos. O sistema permite imprimir ou exportar a estrutura dos gráficos, além dos logs correspondentes aos sensores selecionados.

### 5.3 Principais resultados

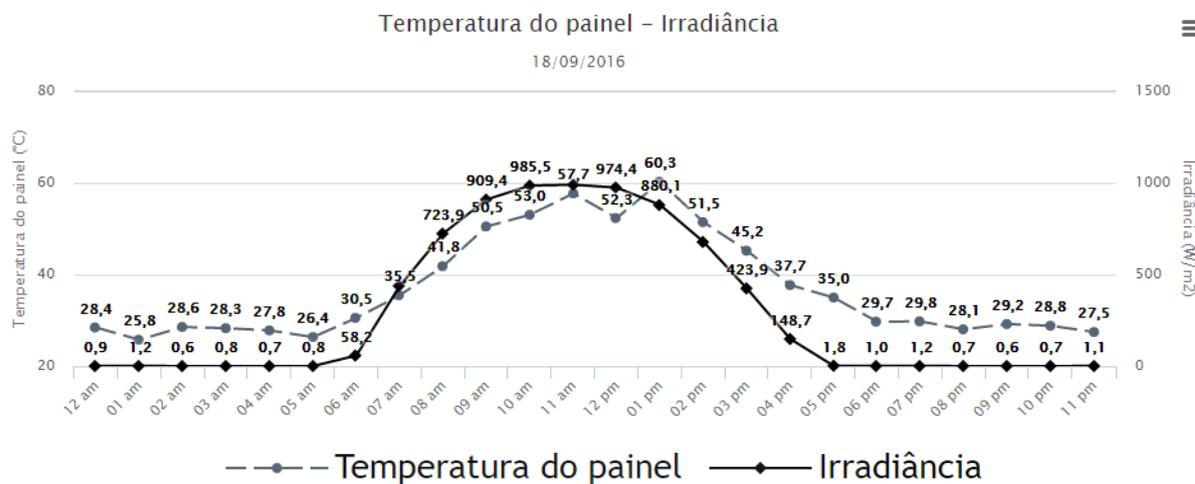
Neste tópico são apresentados os resultados obtidos com a aplicação Web para monitoramento *online* referente aos dados de geração de um painel FV alimentando uma planta de bombeamento de água, com os respectivos gráficos das variáveis monitoradas.

#### 5.3.1 Comportamento da temperatura do painel FV em relação à irradiância

A Figura 39 apresenta o comportamento da temperatura do painel FV e a irradiância no dia 18 de setembro de 2016. Como esperado, a temperatura do painel aumenta de acordo com a irradiância solar, que pode chegar até 1000 W/m<sup>2</sup>. Em relação à temperatura do painel FV, o valor máximo de 60 °C é atingido em torno do meio dia, enquanto a temperatura ambiente apresenta uma média de 31 °C. A umidade do ar manteve-se em uma média de 60 % durante o mês de setembro. Apesar da cidade de Fortaleza apresentar um bom potencial solar, apresenta também altas temperaturas, o que reduz a eficiência do painel. A irradiação solar diária deste dia foi de 7,12 kWh/m<sup>2</sup>. A Figura 40 apresenta as medições de

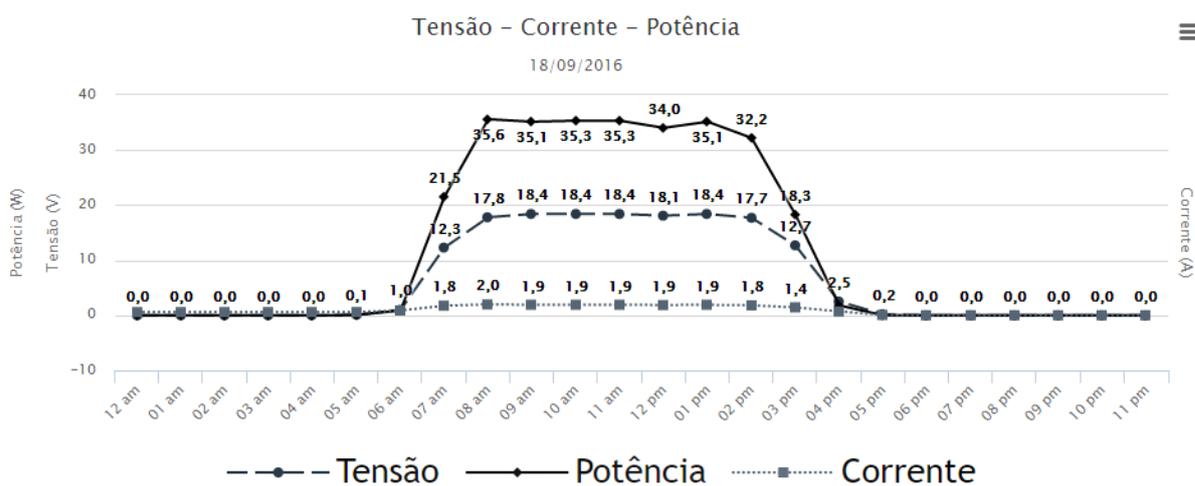
tensão, corrente e potência do painel FV no mesmo dia. Como esperado, em torno do meio-dia tem-se o máximo valor de potência, com tensão chegando a 18,4 V, corrente de 1,92 A e potência máxima de 36 W.

Figura 39 - Temperatura do painel FV e irradiância solar no dia 18/09/2016.



Fonte: Elaborado pelo autor.

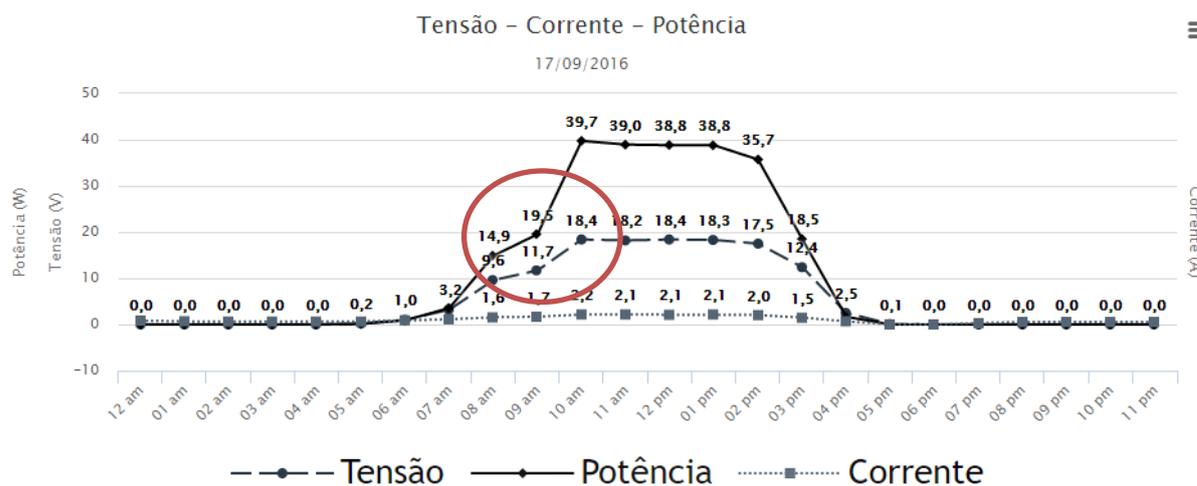
Figura 40 - Tensão, corrente e potência do painel FV no dia 18/09/2016.



Fonte: Elaborado pelo autor.

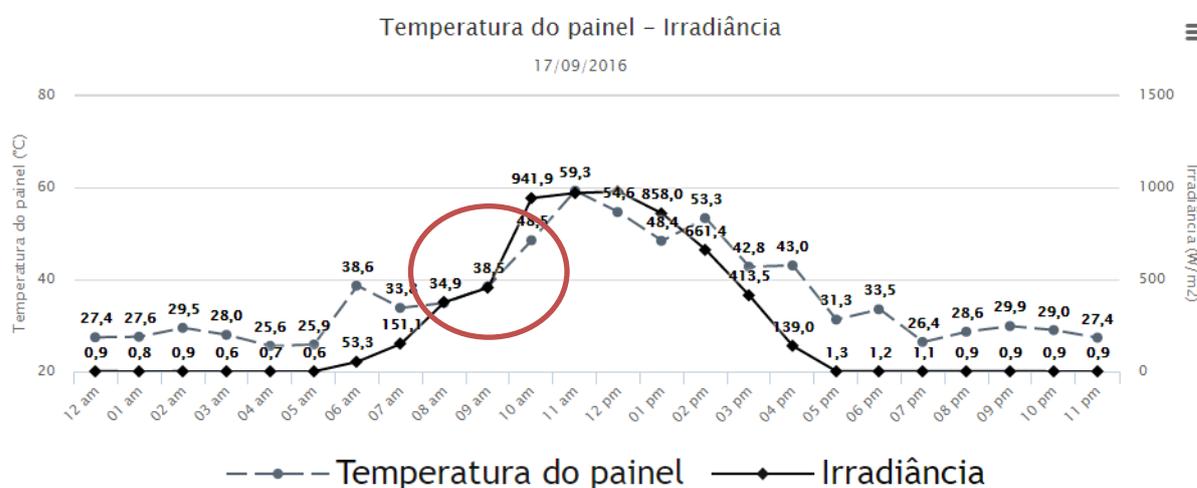
A Figura 41 e a Figura 42 mostram os gráficos de tensão, corrente e potência elétrica e temperatura do painel FV e irradiância solar no dia 17 de setembro de 2016, respectivamente, extraídos da aplicação Web. Observa-se uma queda na geração no horário de 07 h da manhã até às 10 h, por céu nublado ou com algumas nuvens intermitentes o que afetou o processo de geração de eletricidade. Neste dia obteve-se uma irradiação de 5,92 kWh/m<sup>2</sup>.

Figura 41 - Tensão, corrente e potência do painel FV no dia 17/09/2016.



Fonte: Elaborado pelo autor.

Figura 42 - Temperatura do painel FV e irradiância solar no dia 17/09/2016.



Fonte: Elaborado pelo autor.

### 5.3.2 Relação entre vazão e irradiância solar

O perfil da vazão do conjunto motobomba no dia 14 de setembro de 2016 em relação à variação da irradiância é ilustrado na Figura 43. Observa-se que o comportamento diário da planta de bombeamento com respeito à vazão da água confrontando-se com a irradiância é similar, indicando a forte dependência da vazão com a variabilidade da irradiância. O bombeamento começou às 07 h até as 16 h, com um volume médio de água bombeado de 2820 litros diários e vazão diária média de 3,67 L/min para uma irradiação solar diária de 6,92 kWh/m<sup>2</sup>. O mesmo gráfico foi levantado em um dia parcialmente nublado (16 de setembro de 2016), conforme a Figura 44. São verificadas reduções nos níveis de vazão

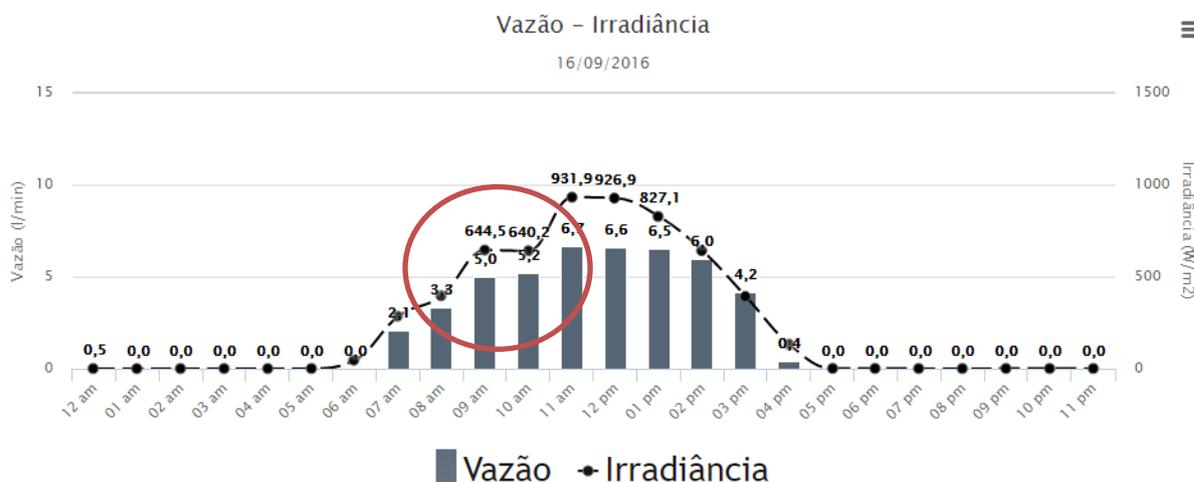
devido à presença de quantidade significativa de nuvens, com irradiação solar diária disponível de 5,79 kWh/m<sup>2</sup>. O sistema foi capaz de bombear em torno de 2520 litros de água.

Figura 43 - Vazão e irradiância no dia 14/09/2016.



Fonte: Elaborado pelo autor.

Figura 44 - Vazão e irradiância no dia 16/09/2016.



Fonte: Elaborado pelo autor.

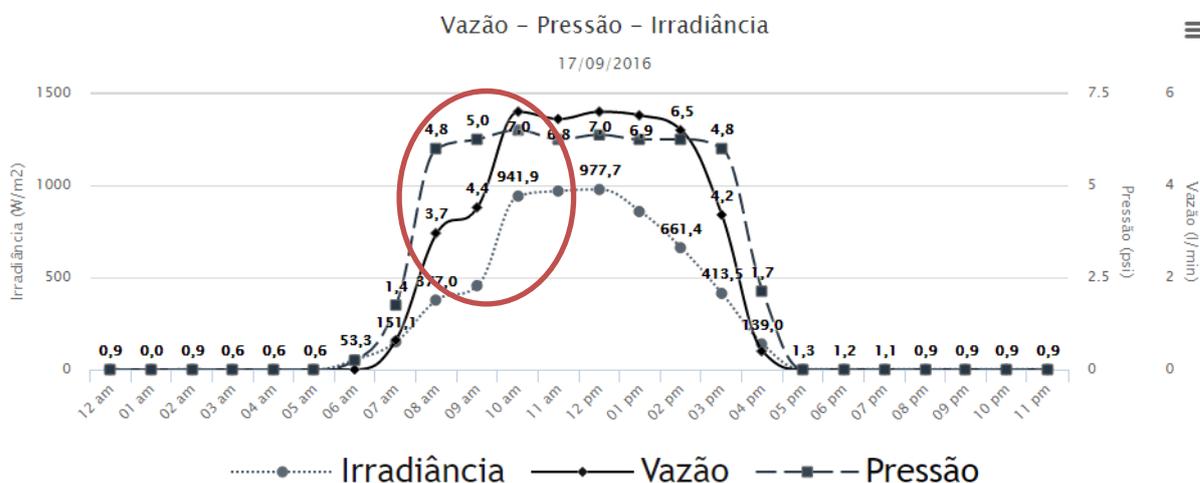
Nos sistemas de bombeamento FV existe uma relação direta entre a irradiância solar e o volume de água bombeada. No dia 16 de setembro de 2016 houve uma queda no volume de água bombeado em virtude da baixa irradiação incidente. Nos dias em que houve aumento da irradiação solar também houve aumento no volume de água bombeado, como ilustrado no dia 14 de setembro de 2016.

### 5.3.3 Relação entre vazão, pressão e irradiância

A vazão e a pressão fornecidas pelo sistema de bombeamento dependem de que a irradiância seja suficiente para o acionamento do conjunto motobomba. Os resultados

referentes à vazão e pressão vinculados à irradiância fornecidas pelo sistema na análise são apresentados na Figura 45 para o dia 17 de setembro de 2016. A irradiação solar diária foi de  $5,92 \text{ kWh/m}^2$  com vazão média ao longo do dia de  $3,40 \text{ L/min}$  e volume diário de água bombeado de 2580 litros com pressão média de  $3,08 \text{ psi}$  na saída da motobomba.

Figura 45 - Vazão, pressão e irradiância no dia 17/09/2016.

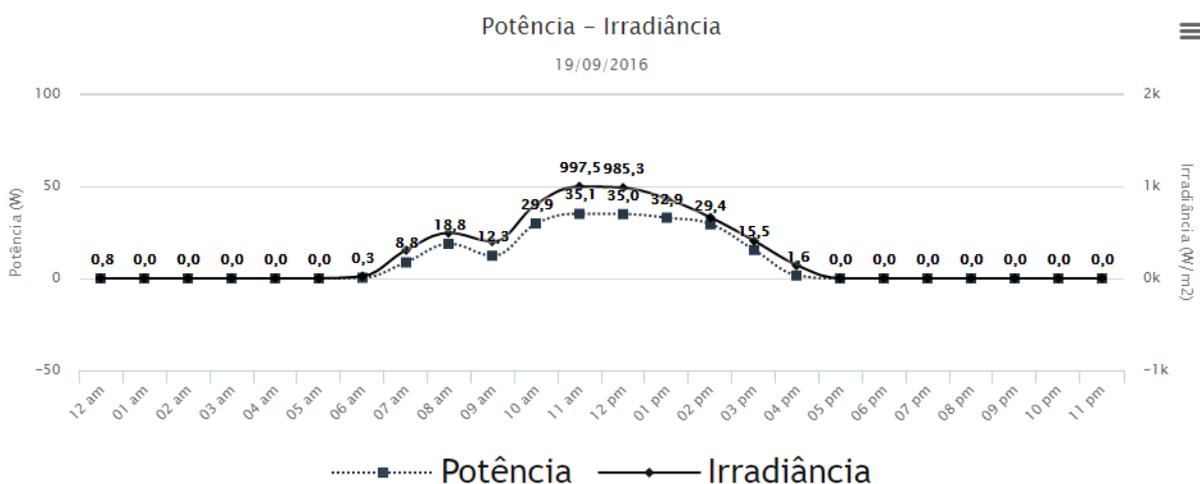


Fonte: Elaborado pelo autor.

### 5.3.4 Relação entre potência elétrica FV e irradiância

A Figura 46 representa a variação da potência elétrica fornecida pelo painel FV em função da irradiância solar no dia 19 de setembro de 2016. Como esperado, a potência FV está relacionada à irradiância disponível. A irradiação solar diária disponível ao longo do dia foi de  $6,02 \text{ kWh/m}^2$  com uma potência máxima de  $35,1 \text{ W}$ .

Figura 46 - Potência elétrica FV em função da irradiância no dia 19/09/2016.



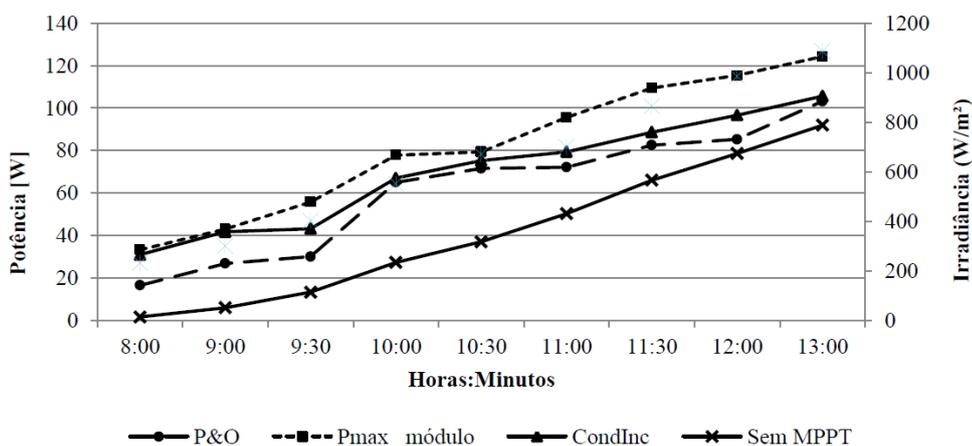
Fonte: Elaborado pelo autor.

### 5.3.5 Eficiência do painel FV, do conjunto motobomba e do sistema geral.

Com finalidade de analisar e avaliar o desempenho de conversão da irradiação solar em energia elétrica, foram calculados os valores diários da corrente e tensão gerados pelo módulo FV e a irradiação diária a partir dos valores coletados a cada minuto no horário das 5 h até as 18 h. A Tabela 10 apresenta os valores calculados da eficiência do painel FV no período de medição de 14 a 23 de setembro de 2016. Observa-se que o módulo apresentou uma baixa eficiência diária em comparação com o dado fornecido pelo fabricante sob condições padrão de teste (eficiência de 14,3 % para irradiação solar de 1000 W/m<sup>2</sup>, temperatura das células de 25 °C e distribuição espectral AM=1,5), uma vez que este comportamento está relacionado diretamente à variação da irradiação solar diária, temperatura de operação e inclinação do módulo FV.

Destacar que neste trabalho não foi utilizado um mecanismo de Seguimento do Ponto de Potência Máxima (*MPPT - Maximum Power Point Tracking*). O conjunto motobomba foi acoplado diretamente ao módulo FV pelo que o ponto de operação não coincidiu com o ponto de máxima potência ( $P_{MP}$  - *Maximum Power Point*). Este ponto varia ao longo do dia, dependendo das condições de irradiação e da temperatura às quais for exposto. (MENDONÇA, 2016) apresenta comparações entre diversas técnicas de *MPPT* e tem como resultado uma redução da potência gerada pelo painel FV sem o uso do *MPPT*. A Figura 47 apresenta uma comparação entre os métodos Perturbar e Observar (P&O) e Condutância Incremental (CondInc) em relação a potência medida sem *MPPT*.

Figura 47 - Potência máxima do módulo FV e dos métodos P&O, CondInc, Sem MPPT



Fonte: (MENDONÇA, 2016).

Outro aspecto que influencia na baixa eficiência é o acúmulo de poeira sobre os módulos pelo que no dia 16 de setembro de 2016 foi realizada uma limpeza do painel FV. Quando comparado com os dias anteriores, observa-se um aumento na eficiência do painel depois do processo de limpeza. Os valores obtidos de eficiência da motobomba e de eficiência global do sistema no mesmo período de medição encontram-se resumidos na Tabela 10. Observa-se que a eficiência do conjunto motobomba fica em torno de 9,74 % em função da tensão e da corrente geradas durante a operação do sistema. A eficiência global do sistema FV de bombeamento foi calculada pela divisão da potência hidráulica pela potência solar.

Tabela 10 - Eficiência calculada pelo SAD desenvolvido

<b>Data da medida</b>	<b>Eficiência do painel FV (%)</b>	<b>Eficiência da motobomba (%)</b>	<b>Eficiência Global (%)</b>
<b>14/09/2016</b>	3,83	10,35	0,40
<b>15/09/2016</b>	3,28	7,65	0,33
<b>16/09/2016</b>	3,95	9,28	0,46
<b>17/09/2016</b>	4,37	9,62	0,42
<b>18/09/2016</b>	4,37	11,52	0,44
<b>19/09/2016</b>	3,7	9,66	0,42
<b>20/09/2016</b>	4,16	10,53	0,42
<b>21/09/2016</b>	5,79	9,75	0,41
<b>22/09/2016</b>	5,13	9	0,38
<b>23/09/2016</b>	5,13	10,07	0,45

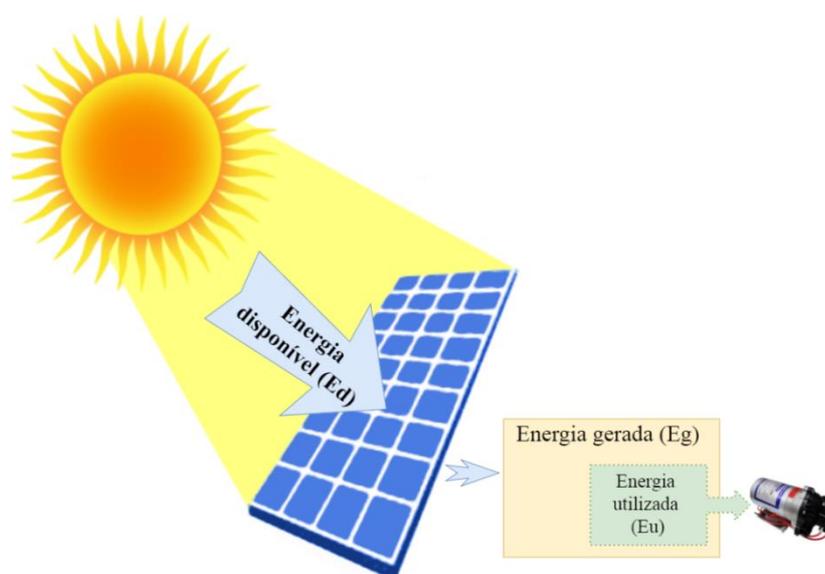
Fonte: Elaborado pelo autor.

Quando comparados com outras pesquisas, os resultados da eficiência global obtidos nesta dissertação se encontram em uma faixa de valores aceitáveis. Por exemplo, em (ARAGAO; SOUZA; ANHALT, 1994), vários sistemas de bombeamento fotovoltaico para garantir o fornecimento de água potável foram instalados em pequenas comunidades rurais na região semiárida do nordeste brasileiro. Como mostrado em (ARAGAO; SOUZA; ANHALT, 1994), o rendimento global do sistema variou entre 1,5 % e 3,5 % mesmo usando tecnologia de ponta.

A energia disponível para o painel FV é fornecida pela irradiação solar sobre a área útil do painel enquanto a energia utilizada pela motobomba é gerada diretamente pelo painel. A quantidade de eletricidade gerada é representada pela área sobre a curva da potência FV fornecida ao longo do dia. A Figura 48 ilustra um diagrama das diferentes energias calculadas no sistema. Com o objetivo de analisar a captação da energia solar e sua conversão em eletricidade, a Tabela 11 apresenta um resumo de vários parâmetros calculados.

O mês de setembro teve dias parcialmente nublados, o que influenciou diretamente no número de horas de funcionamento da planta de bombeamento FV. Entretanto, nos dias de céu limpo, como no dia 18, a irradiação atingiu valor diário em torno de  $7,12 \text{ kWh/m}^2$  e a eficiência de conversão alcançou  $4,37 \%$ ; comparando com um dia com uma quantidade significativa de nuvens no céu, como no dia 19, pode observar-se uma redução em todos os parâmetros calculados. Além da interferência dos elementos climáticos a que o módulo FV é submetido, existem outros como sujeira e poeira que afetam o adequado funcionamento do módulo. Dentro destes elementos, o principal é a temperatura ambiente elevada (faixa entre  $30$  e  $32^\circ\text{C}$ ) a que as células são expostas nesta região. Devido a isto e a não utilização de um mecanismo de *MPPT*, o módulo FV não chegou a atingir a eficiência que o fabricante determina, afetando o processo de conversão de energia, pois estes são testados em ambiente controlado.

Figura 48 - Diagrama das energias calculadas no sistema.



Fonte: Elaborado pelo autor.

Observa-se através do percentual de utilização, como a energia gerada pelo o painel é aproveitada. Neste caso, aproximadamente  $0,54 \%$  da energia gerada é utilizada pelo conjunto motobomba. Note-se que o percentual de utilização é menor do que a eficiência de conversão porque tem horas que o painel fornece, mas a motobomba fica desligada pois a irradiância não é suficiente.

Tabela 11 - Energia utilizada, disponível e gerada pelo modulo FV (setembro de 2016).

Dia	Setembro				
	Energia disponível para o painel FV (Ed) (kWh)	Energia gerada pelo painel FV (Eg) (Wh)	Energia utilizada pela motobomba (Eu) (Wh)	Percentual de utilização (Eu/Eg) (%)	Eficiência de conversão (%)
14	6,92	190,89	102,79	0,54	3,83
15	5,65	131,53	70,82	0,54	3,28
16	5,79	160,04	86,17	0,54	3,95
17	5,92	183,25	98,67	0,54	4,37
18	7,12	220,25	118,6	0,54	4,37
19	6,02	157,6	84,86	0,54	3,7
20	6,58	189,95	102,28	0,54	4,16
21	6,06	241,32	129,94	0,54	5,79
22	5,74	202,57	109,08	0,54	5,13
23	6,06	214,84	115,68	0,54	5,13

Fonte: Elaborado pelo autor.

#### 5.4 Comparação entre medições

A etapa de comparação entre medições do sistema desenvolvido na presente dissertação e medições de um sistema comercial faz-se necessária para garantir a confiabilidade dos dados coletados. Assim, são comparadas as medições do SAD desenvolvido com as de um sistema supervisorio constituído por um Controlador Lógico Programável (CLP) utilizado em pesquisas anteriores do LEA-UFC e que apresentam boa precisão, a fim de que possam ser identificadas falhas quanto à precisão das medições, permitindo realizar ajustes no sistema, caso necessário.

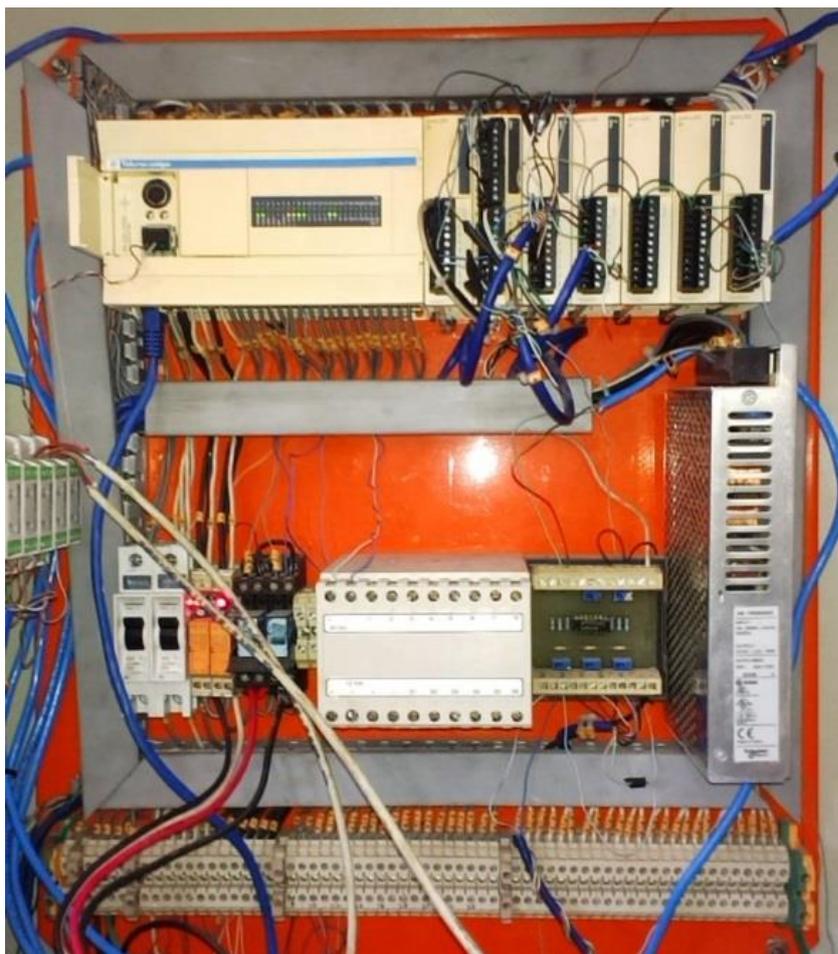
O sistema de supervisão e aquisição de dados comercial utilizado está composto por um *software* para monitorar, supervisionar e armazenar as variáveis e os dispositivos de sistemas de controle conectados através de drivers específicos, utilizando controladores lógicos programáveis (CLP) como protocolo de comunicação. A programação do CLP é efetuada pelo *software TwidoSuite V2.20*, sendo a interface com o computador efetuada através do *Driver Modicon Modbus*, que serve para implementar o protocolo de comunicação. O CLP faz aquisição de dados a cada minuto, onde as informações depois de coletadas são exportadas para um arquivo em Excel (MENDONÇA, 2016).

Todas as medições efetuadas pelos sensores descritos são coletadas por um sistema supervisorio composto por controladores lógicos programáveis, ilustrado na Figura 49. A seguir é apresentado o hardware e o software do Sistema supervisorio para coletar e

armazenar os dados da planta de bombeamento fotovoltaico. As informações dos sensores configurados no CLP são capturadas e armazenadas por um aplicativo desenvolvido utilizando o software ELIPSE SCADA, conforme tela apresentada na Figura 50.

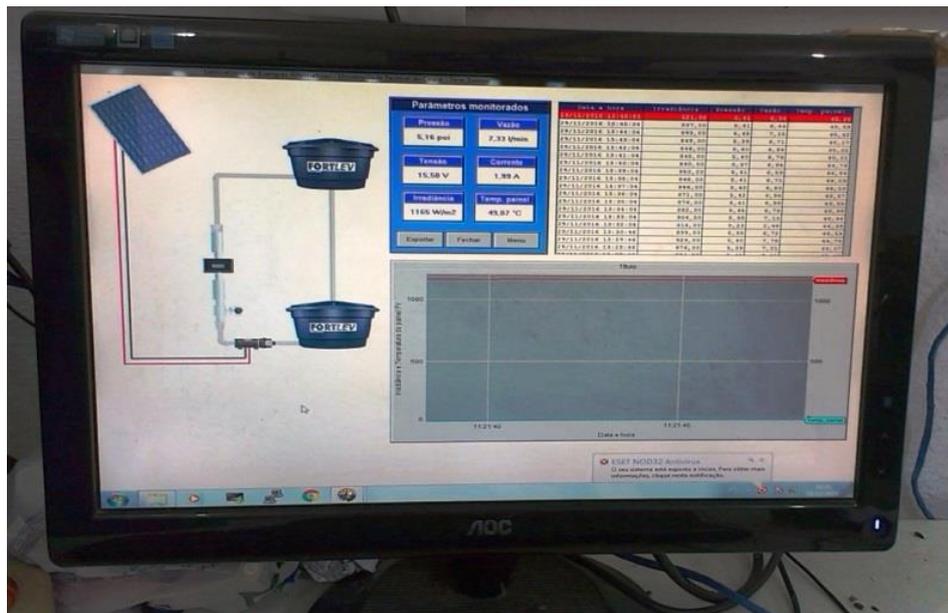
Este aplicativo permite coletar dados de tensão, temperatura e corrente do painel FV, irradiância global e pressão e vazão de entrada no conjunto motobomba. Depois de capturadas, todas as informações são exportadas para um arquivo em Excel, podendo ser trabalhadas e devidamente analisadas.

Figura 49 - Sistema supervisorio para coletar e armazenar dados.



Fonte: Elaborado pelo autor.

Figura 50 - Tela do aplicativo desenvolvido para captura de dados

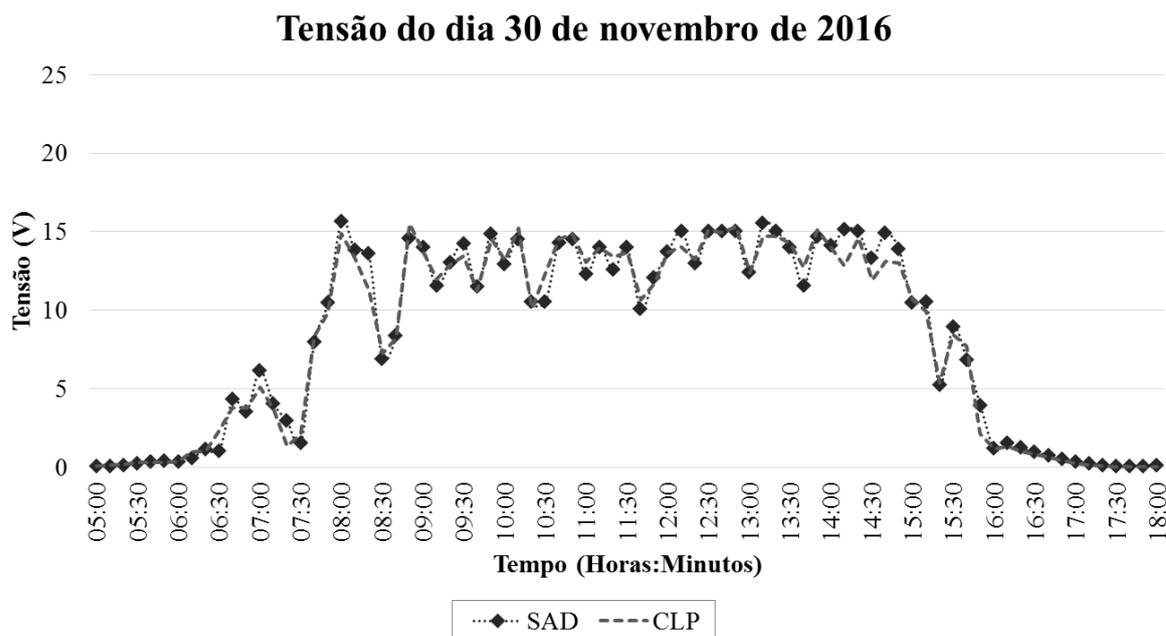


Fonte: Elaborado pelo autor.

Para a validação dos dados de todos os sensores configurados no SAD desenvolvido, foram comparadas as medições com as mesmas informações coletadas pelo SAD comercial (CLP). A comparação foi feita em intervalos de 10 minutos após exportar os dados do SAD e do CLP para um arquivo Excel. Isto permitiu desenhar e analisar os gráficos eliminando os efeitos das variações abruptas, provocadas pelas variações na irradiância. As figuras seguintes mostram os resultados da comparação entre os dois sistemas (SAD e CLP) no dia 30 de novembro de 2016 para todas as variáveis monitoradas e do dia 31 de dezembro de 2016 para a temperatura do painel FV. Pode ser visto que todas as curvas apresentadas seguem um padrão semelhante, indicando boa precisão pelo sistema desenvolvido (SAD).

A Figura 51 apresenta a comparação entre os dois sistemas em relação à tensão de saída do módulo FV. A tensão média diária medida pelo SAD foi 8,34 V, enquanto a mesma variável medida pelo CLP retornou uma tensão média diária 8,21 V. O erro do cálculo no período medido pelo SAD em relação ao CLP foi de 1,52 %.

Figura 51 - Comparação da tensão do painel FV entre SAD e CLP (30/11/2016).

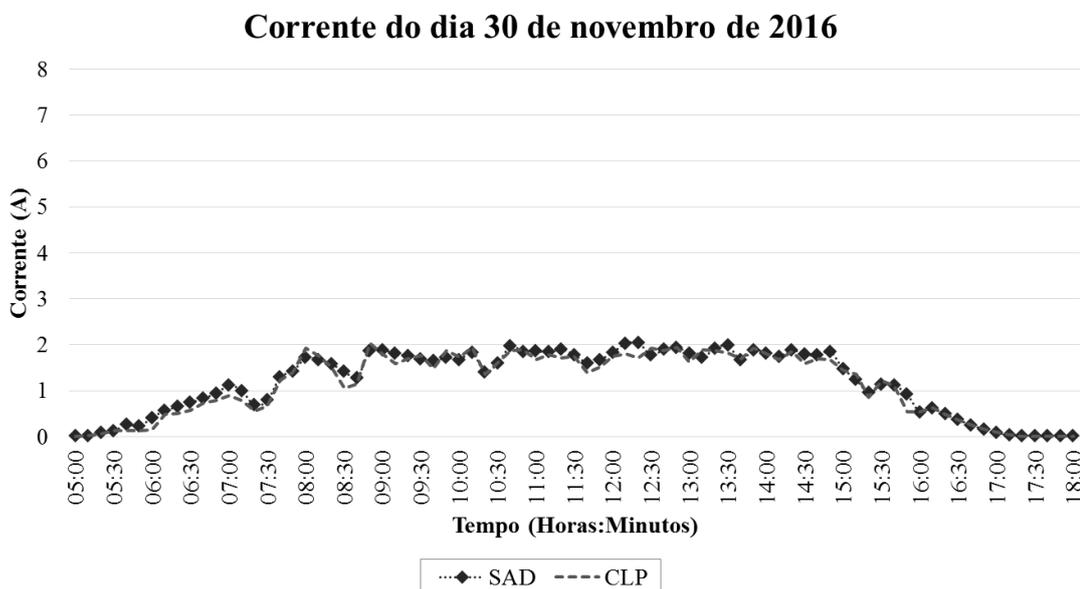


Fonte: Elaborado pelo autor.

### A

Figura 52 mostra a comparação entre os dois sistemas em relação à corrente gerada pelo módulo FV. A corrente média diária medida pelo SAD foi de 1,20 A, enquanto a mesma variável medida pelo CLP retornou uma corrente média diária de 1,14 A. O erro do cálculo no período medido pelo SAD em relação ao CLP foi de 5,18 %.

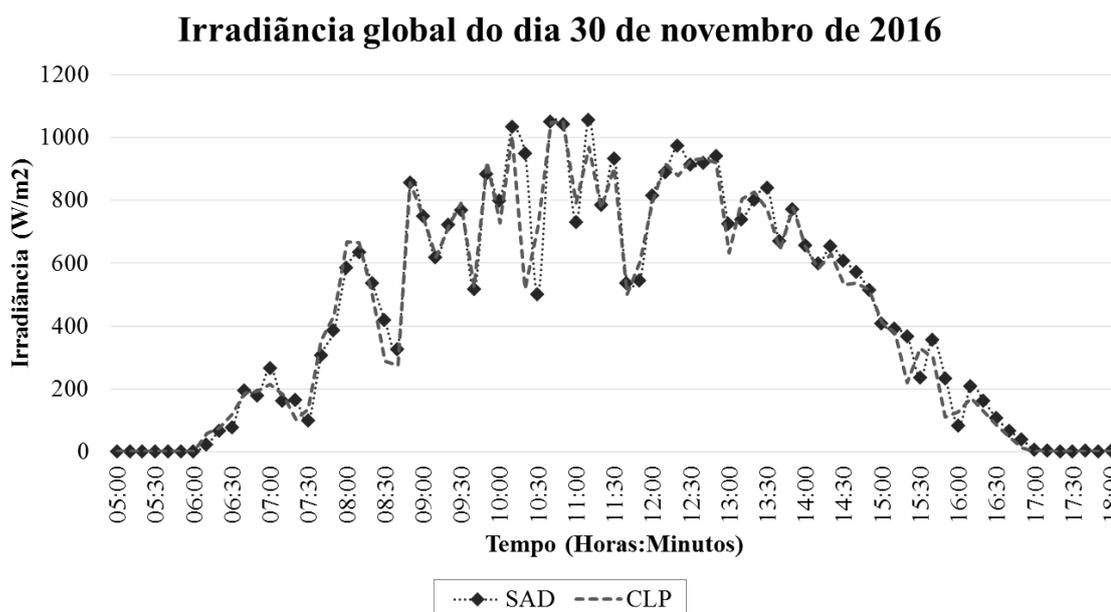
Figura 52 - Comparação da corrente gerada pelo painel FV entre SAD e CLP (30/11/2016).



Fonte: Elaborado pelo autor.

A Figura 53 apresenta a comparação entre os dois sistemas em relação à irradiância global diária disponível para o módulo fotovoltaico. A irradiância média diária medida pelo SAD foi  $5,88 \text{ kWh/m}^2$ , enquanto a mesma variável medida pelo CLP retornou uma irradiância média diária de  $5,73 \text{ kWh/m}^2$ . O erro do cálculo no período medido pelo SAD em relação ao CLP foi de 2,53 %.

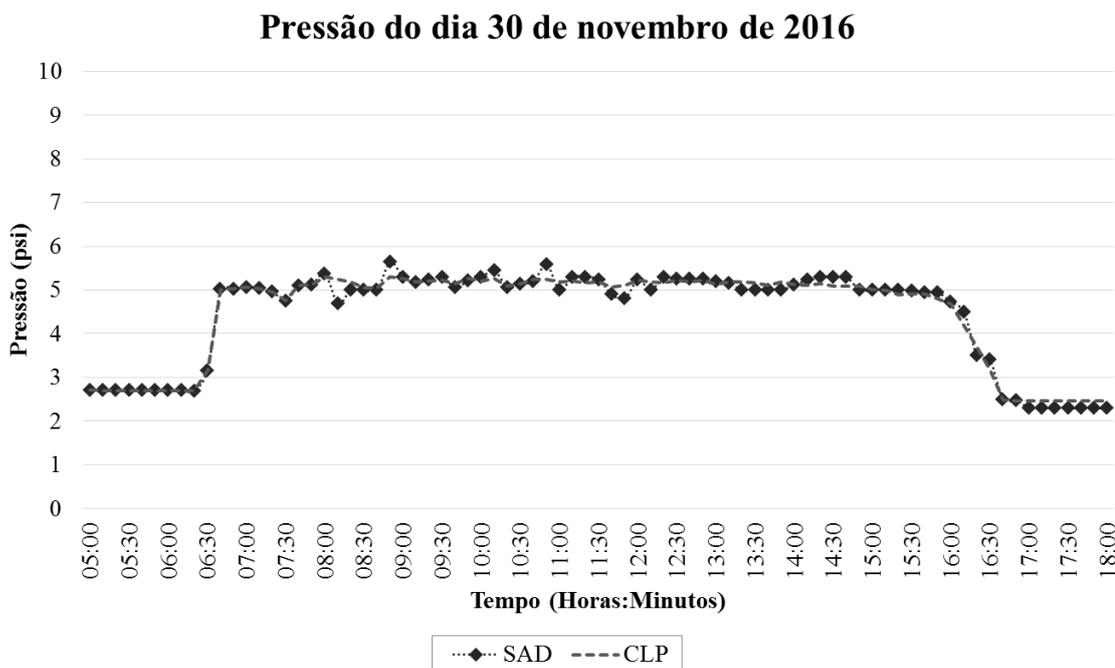
Figura 53 - Comparação da irradiância solar disponível entre SAD e CLP (30/11/2016).



Fonte: Elaborado pelo autor.

A Figura 54 mostra a comparação entre os dois sistemas em relação a pressão do conjunto do motobomba. A pressão média diária medida pelo SAD de 4,45 psi enquanto a mesma variável medida pelo CLP retornou uma pressão média diária de 4,46 psi para um erro de medição de 0,16 %.

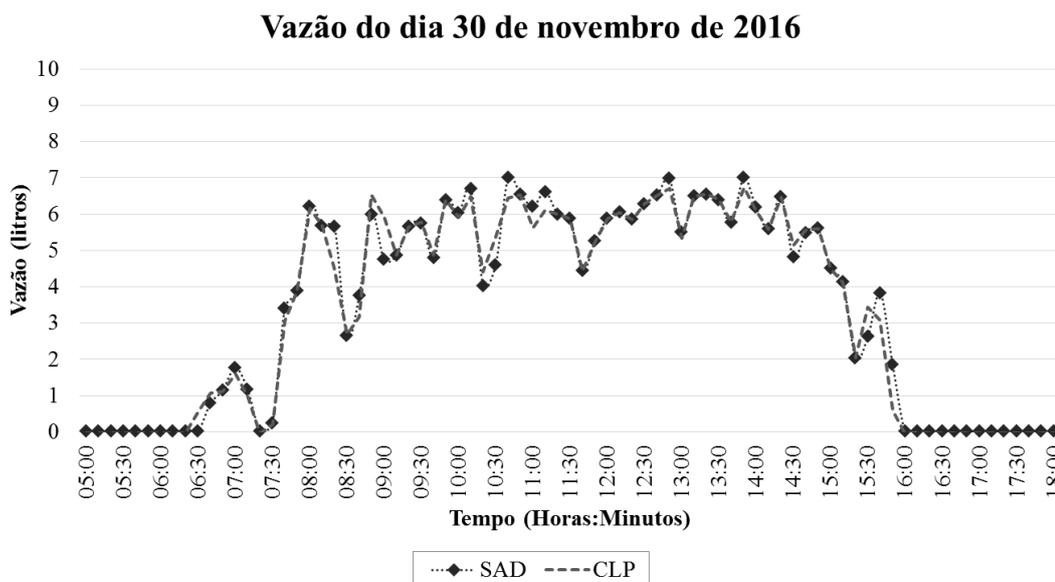
Figura 54 - Comparação da pressão entre SAD e CLP (30/11/2016).



Fonte: Elaborado pelo autor.

A Figura 55 apresenta a comparação entre os dois sistemas em relação a vazão de água diária. A vazão diária medida pelo SAD foi de 3,45 L/min, enquanto a mesma variável medida pelo CLP retornou uma vazão média diária de 3,41 L/min para um erro de medição de 1,25 %.

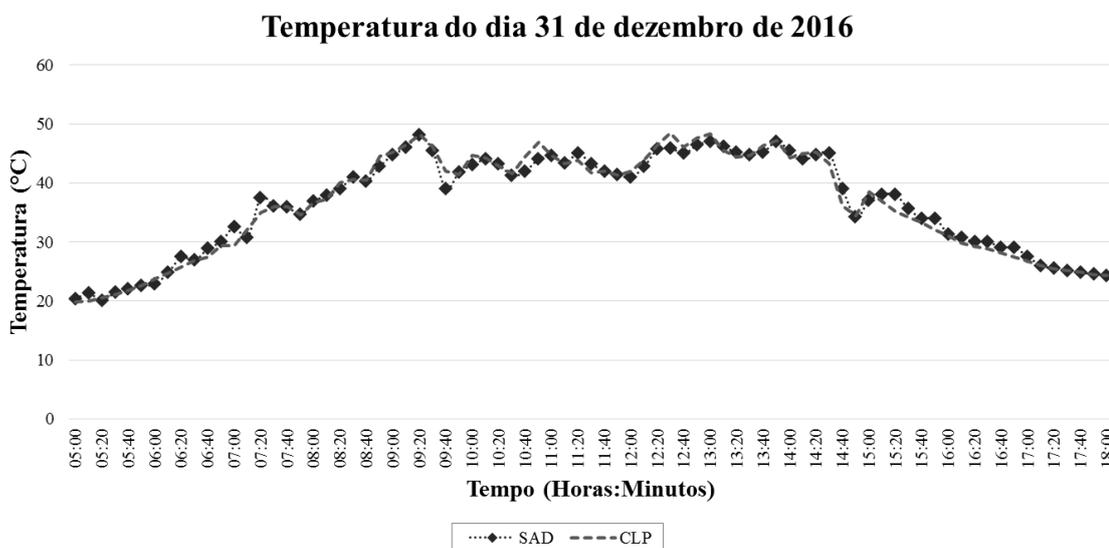
Figura 55 - Comparação da vazão entre SAD e CLP (30/11/2016).



Fonte: Elaborado pelo autor.

A Figura 56 apresenta a comparação entre os dois sistemas em relação à temperatura do painel FV. À temperatura média diária medida pelo SAD foi de 36,52 °C, enquanto a mesma variável medida pelo CLP retornou uma temperatura média diária de 36,44 °C para um erro de medição de 0,21 %.

Figura 56 - Comparação da temperatura do painel FV entre SAD e CLP (31/12/2016).



Fonte: Elaborado pelo autor.

#### ***5.4.1 Análise dos sistemas de aquisição comparados***

Esta seção descreve os diferentes aspectos avaliados nos dois SADs comparados.

A

Tabela 12 apresenta um quadro comparativo com os diferentes resultados. Em relação ao preço, o SAD desenvolvido foi implementado com componentes de baixo custo adquiridos no mercado brasileiro enquanto o SAD comercial apresenta custos relativamente elevados. O aspecto da portabilidade está relacionado com a possibilidade de transporte para outros locais ou uso em um computador portátil. Em relação a este aspecto, o SAD desenvolvido possui um tamanho de dispositivo reduzido e sua conexão com uma porta de comunicação padrão do computador (USB) e possibilidade de uso de redes sem fio são determinantes para utilização em diversos ambientes. O SAD comercial baseado em CLP não é adequado para tais experiências. Seu pequeno tamanho e capacidade compacta faz ocupar pouco espaço físico, sendo fácil de instalar e programar e requer pouca manutenção. O SAD desenvolvido utiliza código aberto e software livre para fazer a mesma tarefa. Oferece a capacidade de armazenar diversas versões dos programas para uso posterior e a possibilidade de modificação ou alteração dos mesmos. O SAD comercial apresenta um grau de dificuldade para programar o hardware para adquirir os dados da forma desejada. Os dados coletados pelo SAD desenvolvidos estão disponíveis na Internet e podem ser acessados desde qualquer dispositivo através de um navegador Web enquanto que a informação coletada pelo SAD comercial, depois de capturadas, tem que ser exportadas para um arquivo em Excel para o posterior tratamento e análise.

Na maioria das comparações, o SAD desenvolvido apresentou um nível de precisão nas medições aceitável quando comparado com o SAD comercial. Durante o decurso da pesquisa foram encontrados erros (duplicação de dados ou programação das tarefas automatizadas) que foram resolvidos com os ajustes necessários, chegando a erros de medição entorno de 3 %.

É certo que os SADs comerciais são muito poderosos na aquisição de grandes quantidades de dados de forma rápida e com muita precisão, porém, o SAD desenvolvido revelou-se muito fácil de usar e com rapidez suficiente para experimentos com uma planta de bombeamento fotovoltaico; apresentando uma capacidade de adaptação das suas funcionalidades a diversas áreas.

Tabela 12 - Análise comparativa dos sistemas de aquisição de dados estudados.

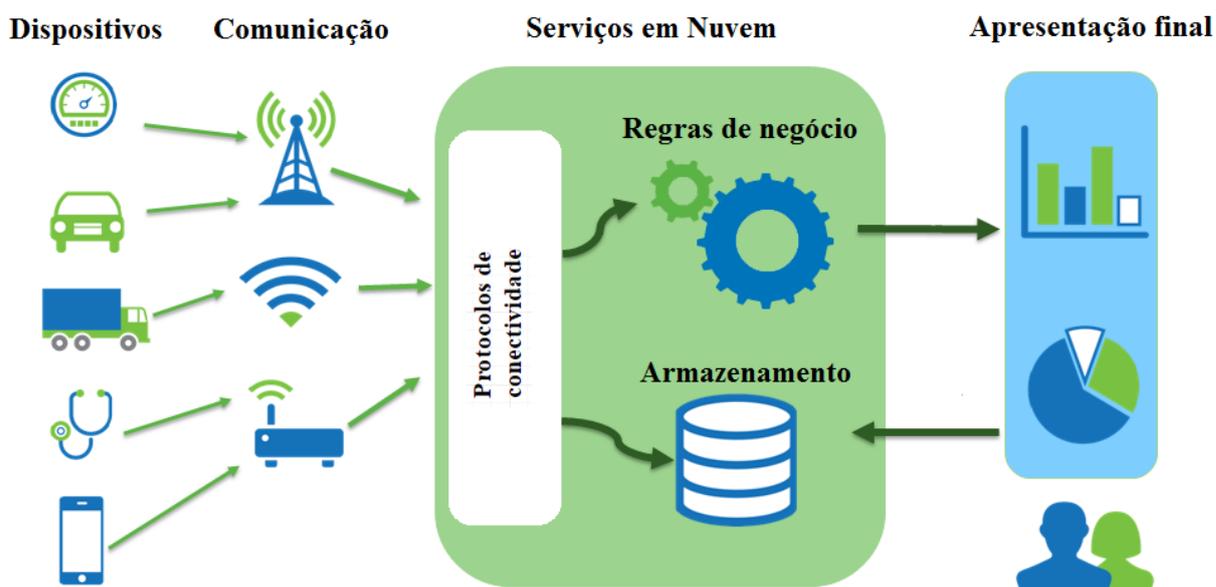
<b>Parâmetro</b>	<b>SAD desenvolvido</b>	<b>SAD comercial</b>
<b>Preço</b>	Baixo custo	Custo elevado
<b>Portabilidade</b>	Sim	Não
<b>Acessibilidade dos dados</b>	Através da Internet	Arquivo Excel
<b>Precisão da medição</b>	Aceitável	Bom

Fonte: Elaborado pelo autor.

### 5.5 Interconexão da planta de bombeamento fotovoltaico à rede IoT

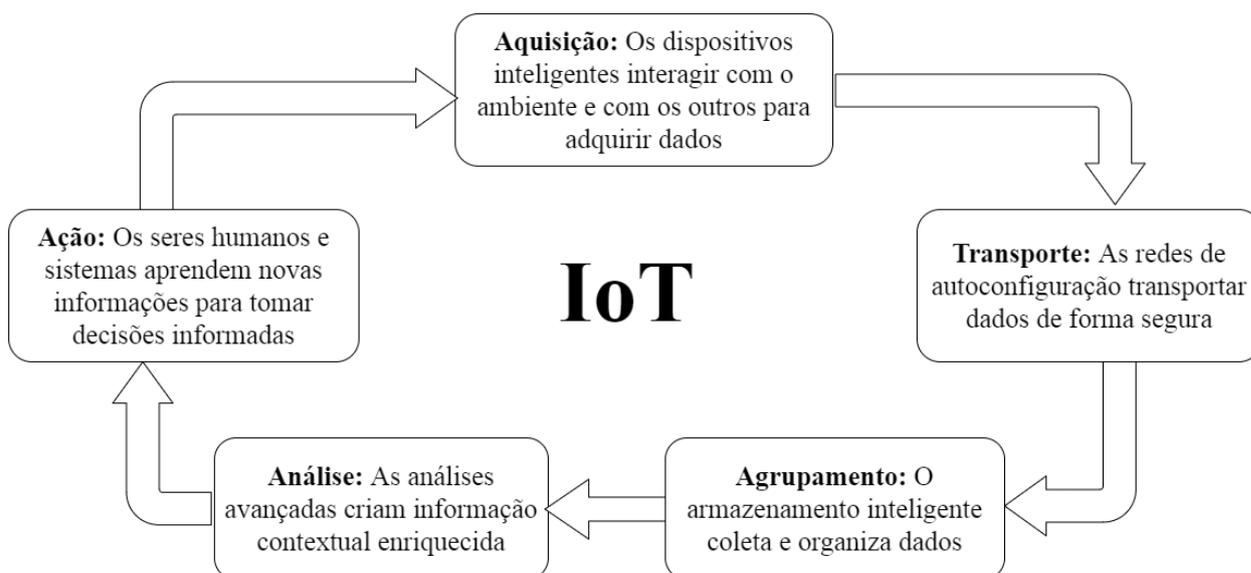
Internet das Coisas (IoT - Internet of Things) é uma rede global de computadores, sensores e atuadores conectados por meio de protocolos de Internet. Em um sentido mais técnico, o que leva à IoT é a integração de sensores e dispositivos em objetos cotidianos que são ligados à Internet através de redes cabeadas ou sem fio, como mostrado na Figura 57. Examinando IoT, se apresentam cinco elementos funcionais de um ciclo lógico de dados: aquisição, transporte, agrupamento, análise e ação, ilustrados na Figura 58. Depois, podemos sobrepor à segurança, governação e administração nesse ciclo, evoluindo para descrever os caminhos para a ação global.

Figura 57 - Internet das Coisas



Fonte: Elaborado pelo autor.

Figura 58 - Elementos do ciclo de dados da Internet das Coisas



Fonte: Elaborado pelo autor.

A aquisição de dados inclui *hardware*, dispositivos inteligentes que coletam dados a partir de interações com o meio ambiente, com outras máquinas, com os seres humanos ou outros seres vivos e torná-los disponíveis para serem transmitidos através de uma rede. Os sensores são as terminações nervosas da *IoT* e os dispositivos de recolhem os dados. Após coleta de dados a partir de diferentes sensores na fase de aquisição de dados, um sistema inteligente deve transportar os dados a uma rede para o agrupamento e análise da informação. As redes cabeadas e sem fio são essenciais para a interligação dos diferentes sensores com o ecossistema mais amplo da *IoT*. Soluções de agrupamento de dados podem incluir bancos de dados, serviços e análises no local, serviços em nuvem ou repositórios de terceiros. A análise de grandes volumes de dados em tempo real é fundamental para ter uma diferenciação e vantagem competitiva nos projetos *IoT*. A função de análise de um sistema *IoT* deve ser capaz de coletar dados de fontes automaticamente, avaliar e produzir uma decisão de acordo com modelos preditivos. A informação e análise para apoiar decisões rápidas e baseadas em dados, estabelecem a base para ações mais complexas e iniciadas por sistemas de otimização de processos e automação operacional. Os sistemas podem assimilar dados do entorno e aprender a melhorar o próprio desempenho.

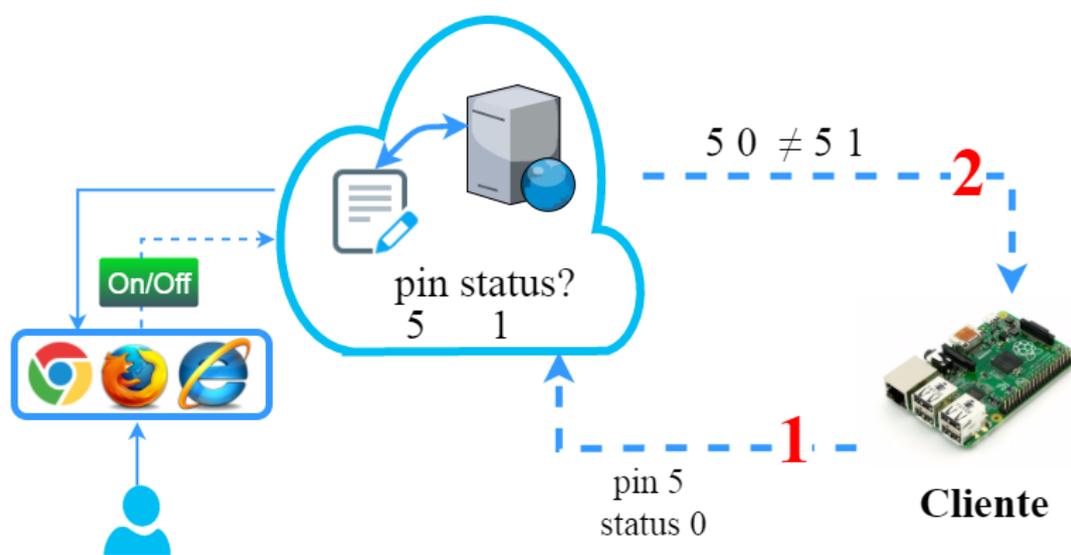
Há diversas formas em que um dispositivo pode se comunicar com outro computador na Web:

- O dispositivo é o cliente que envia ou recebe dados para um servidor.

- O dispositivo é o servidor, que fornece ou aceita dados dos clientes.

Com o objetivo de garantir que esta pesquisa seja aplicável a projetos de *IoT* com acesso remoto via Internet e não somente Intranet, foi desenvolvida uma interface Web para ter acesso ao funcionamento da planta de bombeamento FV e executar ações. Existem duas opções fundamentais para garantir a comunicação entre o servidor e o cliente: *Push* e *Pull*. O primeiro refere-se a um modelo de comunicação entre cliente-servidor onde é o servidor que inicia a comunicação, enquanto o segundo estabelece que os dispositivos vão enviar os dados sem perguntar. Nesta pesquisa foi implementada a opção *Pull* ilustrada na Figura 59 através de um mecanismo de leitura/escrita de arquivos no servidor. Neste caso, o dispositivo vai consultar periodicamente se houve mudanças no índice armazenado no arquivo no servidor Web na Nuvem. Em caso positivo, vai executar o comando solicitado, em caso negativo, vai continuar consultando até haver alterações.

Figura 59 - Implementação do mecanismo *Pull*.



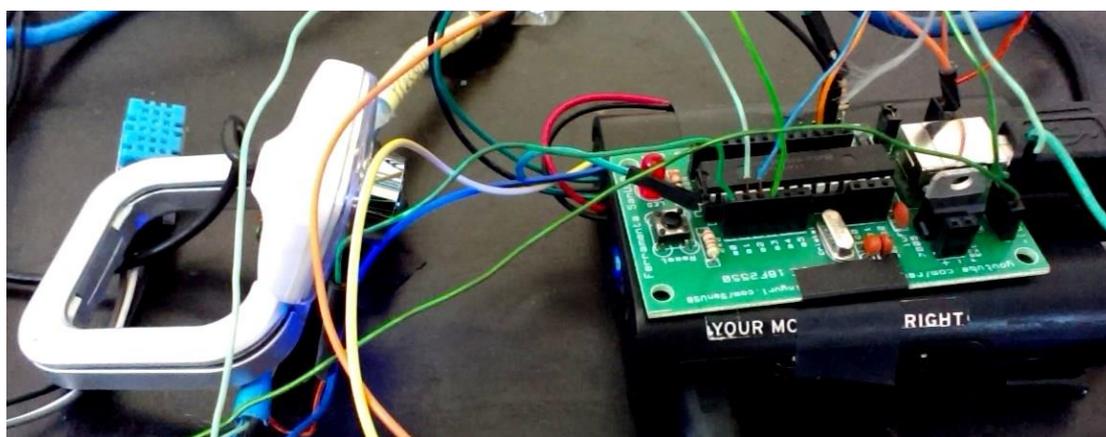
Fonte: Elaborado pelo autor.

Para a maioria, as ferramentas de monitoramento em rede são muitas vezes utilizadas para determinar o estado e funcionamento das comunicações e infraestrutura através de relatórios. No entanto, em princípio, sistemas de monitoramento normalmente não fornecem elementos envolvidos na resolução de problemas localizados. Com a aplicação de novas tecnologias, este problema é resolvido. Com a evolução constante das tecnologias de monitoramento e aquisição de dados utilizando *software* livre, é possível notarmos uma redução considerável na implementação destes sistemas. Com o objetivo de garantir a segurança do ambiente monitorado, resulta uma boa prática a implementação de sistemas de

monitoramento por vídeo aplicados a *IoT*. Estes sistemas surgem a título de complementar o processo tradicional de aquisição de dados, dando ao usuário a capacidade de interagir com o sistema monitorado e a oportunidade de detectar falhas e corrigi-las automaticamente. A Figura 60 apresenta a infraestrutura composta por uma câmera Web conectada via USB ao *Rpi* pela qual será possível visualizar os resultados das ações executadas.

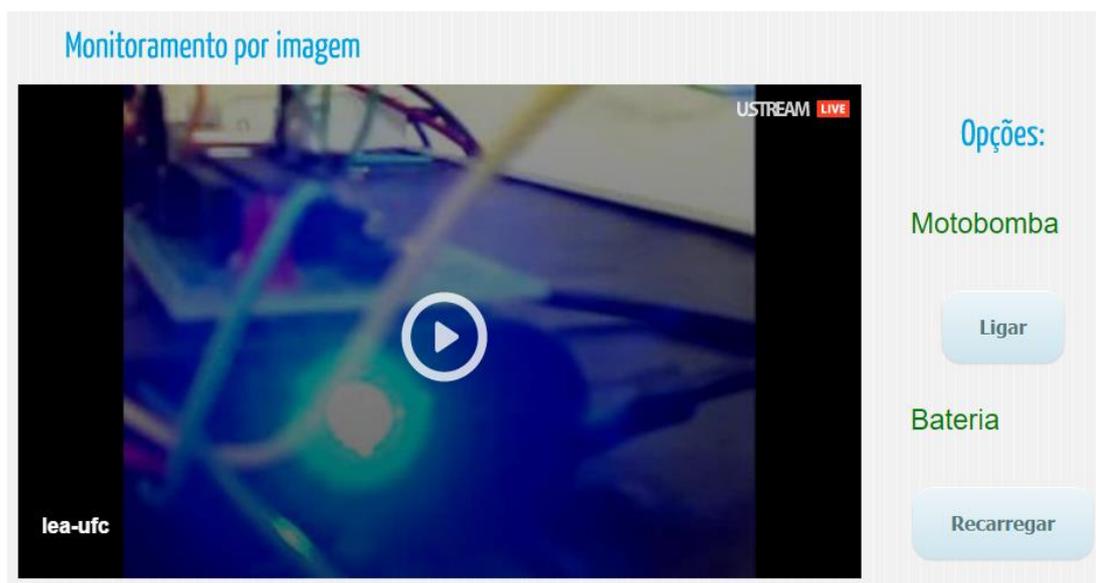
Para a visualização via Web do funcionamento da planta de bombeamento FV em tempo real, foi usado à plataforma online Ustream. Esta plataforma tem vários canais que permitem a transmissão de eventos ao vivo, sem qualquer instalação de software. Para este fim, um canal foi criado nesta plataforma chamado LEA-UFC através do qual pode ser visualizada a operação do sistema de monitoramento e o resultado das ações selecionadas pelos usuários. Com o rápido desenvolvimento da tecnologia de monitoramento por vídeo, são muitas áreas onde podem ser aplicados. Estes sistemas filtram eficazmente uma grande quantidade de dados, rastreiam e analisam objetos em movimento nas fontes de vídeo, detectam comportamentos anormais e geram alarmes para que sejam tomadas as medidas adequadas. Um exemplo de aplicação deste tipo de sistemas de monitoramento por vídeo é apresentado a seguir. Neste caso foi desenvolvida uma interface para o usuário acionar ou não o conjunto motobomba e observar o resultado de suas ações em tempo real a traves da Internet. A Figura 61 mostra uma imagem da placa de comandos fornecida ao usuário. Como ilustrado na imagem, no lado direito se encontram as opções “Motobomba” e “Bateria”. Neste caso, o usuário tem a possibilidade de ligar ou desligar o conjunto motobomba e recarregar ou não uma bateria acoplada, para fornecer outra ação como exemplo.

Figura 60 - Estrutura proposta para o monitoramento via Nuvem



Fonte: Elaborado pelo autor.

Figura 61 - Opções para o acionamento da planta via Internet.



Fonte: Elaborado pelo autor.

Depois de escolhida alguma opção na Web, o usuário pode visualizar uma sinalização através de um *led* que corresponde a uma ação específica. Nesta tela é mostrada uma imagem em tempo real do sistema de semáforo desenvolvido com o objetivo de ter uma fácil interação com o sistema. A opção proposta representa uma interface entre os periféricos - neste caso a motobomba e a bateria - e o microcontrolador através do uso das portas GPIO do Rpi. Como as portas GPIO não têm função definida e por padrão não são usadas, podem ser usadas para prover essa sinalização. Neste caso, são usadas as portas GPIO 4 e 5 para o acionamento da bateria e da motobomba, respectivamente. Os pinos GPIO podem ser ativados ou desativados e configurados para serem entrada ou saída. A Tabela 13 apresenta a cor dos *leds* que representa cada ação a executar e a porta *GPIO* onde estão conectados.

Tabela 13 - Portas GPIO e ações a executar.

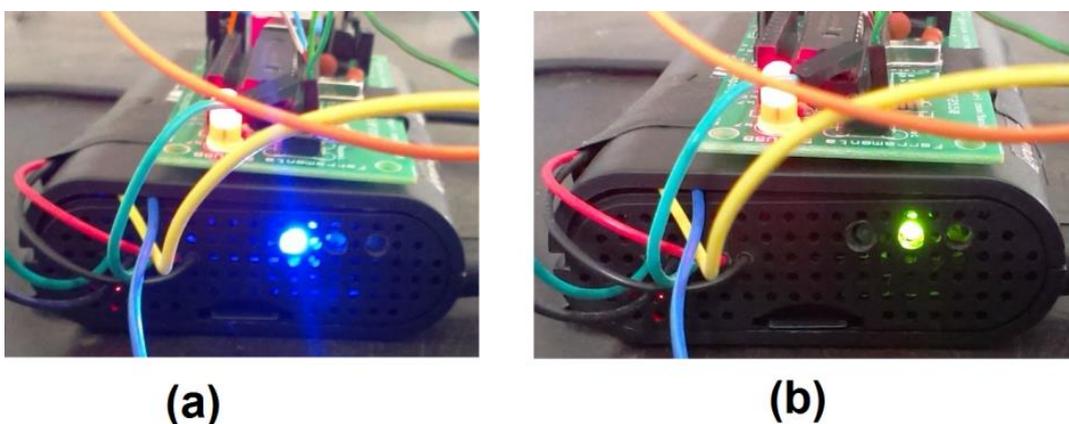
Cor	Objeto	Porta GPIO	Status	Ação
Amarelo	Bateria	2	0	Bateria não recarregando
		2	1	Bateria recarregando
Azul	Motobomba	7	0	Motobomba desligada
		7	1	Motobomba ligada

Fonte: Elaborado pelo autor.

A questão da criação de um aplicativo da Web seguro é muito ampla e requer um estudo para entender as vulnerabilidades de segurança. Neste sentido, devem ser tomadas medidas básicas para proteger qualquer aplicação Web. A principal medida é a execução de aplicativos com privilégios mínimos. Quando o aplicativo é executado, ele está em um contexto que tem privilégios específicos no computador local e, possivelmente em computadores remotos; por isso não deve ser executado com a identidade de um usuário do sistema (administrador). A injeção de código SQL é um ataque muito comum e perigoso na Internet. Com a exploração desta vulnerabilidade, pode ser obtido o controle total sobre o banco de dados ou até mesmo executar comandos no sistema. Para evitar tais riscos e ameaças, é uma boa prática não executar comandos diretamente da Web, de modo que são necessárias alternativas para efetuar a mesma ação.

Neste caso, foi desenvolvido um mecanismo de leitura/escrita que permite a ativação ou desativação dos pinos GPIO do Rpi através da Internet. Esta solução representa uma opção para controlar o Rpi via Web através da manipulação dos pinos de entrada/saída sem necessidade de fazer alterações nos *firewalls* ou abrir portas para permitir o acesso. Foi desenvolvido um programa para ler um arquivo no servidor Web que contém o número da porta selecionada desde a interface Web e o status dela. A Figura 62a apresenta o exemplo de acionamento da motobomba e a Figura 62b representa o acionamento da bateria.

Figura 62 - Acionamento da motobomba (a) e bateria (b).



Fonte: Elaborado pelo autor.

Neste caso, foi escolhida a opção “Ligar a motobomba” o que representa executar o comando `gpio write 5 0` e `gpio write 7 1` para ativar o led azul no Rpi. A Figura 63 ilustra o resultado desta ação.

Figura 63 - Resultado da execução dos comandos no Rpi.

pin=5  
status=0 → Número e estado da porta

BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM
		3.3v			1	2		5v		
2	8	SDA.1	IN	1	3	4		5V		
3	9	SCL.1	IN	1	5	6		0v		
4	7	GPIO. 7	OUT	1	7	8	1	ALT0	15	14
		0v			9	10	1	ALT0	16	15
17	0	GPT0. 0	IN	0	11	12	1	OUT	1	18
27			OUT	0	13	14		0v		
22			IN	0	15	16	1	OUT	4	23
					17	18	0	OUT	5	24
10			IN	0	19	20		0v		
9	13	MISO	OUT	0	21	22	0	IN		
11	14	SCLK	IN	0	23	24	1	IN		
		0v			25	26	1	IN		
0	30	SDA.0	IN	1	27	28	1	IN		
5	21	GPIO.21	IN	1	29	30		0v		
6	22	GPIO.22	IN	1	31	32	0	IN	26	12
13	23	GPIO.23	IN	0	33	34		0v		
19	24	GPIO.24	IN	0	35	36	0	IN	27	16
26	25	GPIO.25	IN	0	37	38	0	IN	28	20
		0v			39	40	0	IN	29	21

Led azul ligado

Motobomba ligada

Fonte: Elaborado pelo autor.

Para assegurar a execução imediata da opção selecionada na web, o programa desenvolvido para esta finalidade foi configurado para ser executado a cada 5 segundos de acordo com a seguinte configuração no Cron do Rpi.

```

* * * * * sh /home/share/Scripts/gpioWeb
* * * * * sleep 5; sh /home/share/Scripts/gpioWeb
* * * * * sleep 10; sh /home/share/Scripts/gpioWeb
* * * * * sleep 15; sh /home/share/Scripts/gpioWeb
* * * * * sleep 20; sh /home/share/Scripts/gpioWeb
* * * * * sleep 25; sh /home/share/Scripts/gpioWeb
* * * * * sleep 30; sh /home/share/Scripts/gpioWeb
* * * * * sleep 35; sh /home/share/Scripts/gpioWeb
* * * * * sleep 40; sh /home/share/Scripts/gpioWeb
* * * * * sleep 45; sh /home/share/Scripts/gpioWeb
* * * * * sleep 50; sh /home/share/Scripts/gpioWeb
* * * * * sleep 55; sh /home/share/Scripts/gpioWeb

```

## **5.6 Conclusões parciais**

O presente capítulo apresentou os principais resultados obtidos com uso do sistema de monitoramento implementado. Conclui-se que apesar dos erros iniciais mostrados nas medições do SAD desenvolvido devido à duplicação de dados e falhas na programação do microcontrolador, e após os ajustes feitos na automatização das tarefas de coleta de dado e na programação do microcontrolador para minimizá-los, foi possível garantir a confiabilidade dos dados coletados.

## 6 CONCLUSÃO

O trabalho foi desenvolvido no Laboratório de Energias Alternativas (LEA) localizado no Campus do PICI da Universidade Federal do Ceará (UFC) em Fortaleza - CE / Brasil. Para o seu desenvolvimento, foi utilizada uma planta de bombeamento FV sem baterias. O SAD desenvolvido permite coletar dados de tensão e corrente FV, irradiância, temperatura ambiente e do painel FV e pressão e vazão do conjunto motobomba. As informações coletadas são tratadas e armazenadas em um servidor Web na Nuvem. Foram usadas ferramentas de *software* livre para o desenvolvimento do sistema a fim de assegurar uma maior divulgação. Dados do funcionamento em tempo real da planta de bombeamento FV podem ser consultados via *smartphones*, *tablets* ou outros dispositivos móveis com acesso à Internet através de um navegador Web sem necessidade da instalação de *software* adicional. O modelo de monitoramento *online* e aquisição de dados proposto pode ser ampliado para armazenar dados de diversos tipos de sensores analógicos ou digitais, bem como de outros tipos de plantas de microgeração utilizando fontes renováveis de energia.

Comparando com as mesmas informações coletadas por um SAD comercial, observa-se que os resultados do SAD desenvolvido apresentam um erro de medição em torno de 3 %. Pode-se concluir então que o sistema desenvolvido apresenta um grau de confiabilidade aceitável dos dados coletados e representa uma solução que pode ser uma proposta competitiva. A primeira versão do SAD desenvolvido foi projetada para coletar e analisar dados de funcionamento em tempo real de uma planta de bombeamento fotovoltaico. Destacar que uma contribuição inovadora desta pesquisa é a possibilidade da aplicação a projetos de *IoT* usando monitoramento por vídeo em Nuvem, conceito pouco abordado nos trabalhos e artigos estudados.

Podem ser citados como principais valores medidos com uso do sistema de aquisição de dados desenvolvido:

- A máxima eficiência obtida para o painel FV foi de 5,13 % em um dia do céu completamente limpo. A irradiação disponível neste dia foi de 6,06 kWh/m<sup>2</sup>.

- O conjunto bomba apresentou uma eficiência máxima de 11,52 %, com o sistema alcançando uma eficiência global máxima de 0,44 %, ambas medidas em condições climáticas favoráveis. A irradiação disponível neste dia foi de 7,12 kWh/m<sup>2</sup>.

- O sistema fornece uma faixa de vazão entre 2,71 e 4,08 L/min de água, bombeando entre 2100 e 3120 litros de água por dia, função do nível de irradiância solar disponível.

### Sugestões para trabalhos futuros

A seguir são citados alguns tópicos como sugestões de continuidade deste trabalho.

- Implementar uma cópia local do site no *Rpi* como redundância em caso de falha da rede.
- Usar um endereço IP real no *Rpi* para ser acessado a partir da Internet.
- Migrar o banco de dados MySQL por *Firebase* oferecido gratuitamente pelo Google. Isso garante uma estrutura mais maleável em formato JSON (*JavaScript Object Notation*) que é um formato de intercâmbio de dados leve, fácil ler e escrever. Assim, pode ser acoplado diretamente para os padrões da *IoT*.
- Utilizar a tecnologia AJAX (*Asynchronous JavaScript and XML*) para substituição das rotinas gráficas criadas pela biblioteca *Highcharts* uma vez que permite enviar e receber informações em uma variedade de formatos, incluindo JSON, XML, HTML, e até mesmo arquivos de texto.
- Implementar o método *Push IoT* garantido menor uso de largura de banda e menor consumo de energia dos dispositivos, pois no método *Pull*, o pedido de transmissão de informação é iniciada pelo cliente.

## REFERÊNCIAS

ADANE, A. et al. Design of a Microcontroller-Based Data Acquisition System for Ground Weather Observations : Evaluation of Radio Refractivity of Air. **International Journal of Communications, Network and System Sciences**, v. 7, p. 355–364, 2014.

ALCANCE LIBRE. **Configuración básica de Apache**. Disponível em: <<http://www.alcanceibre.org/staticpages/index.php/como-apache>>. Acesso em: 11 abr. 2016.

ALLEGRO MICROSYSTEMS. ACS712 Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor. **Datasheet**, p. 1–15, 2013.

ALVARENGA, C. A. **Bombeamento de água com energia solar fotovoltaica** Belo Horizonte Solenerg Engenharia e Comércio Ltda., , 2015.

ANEEL. Energia Solar. **Atlas da Energia Elétrica do Brasil**, p. 14, 2005.

APACHE. **Welcome! - The Apache HTTP Server Project**. Disponível em: <<https://httpd.apache.org/>>. Acesso em: 11 abr. 2016.

ARAGAO, P. M. C.; SOUZA, A. DE C.; ANHALT, J. Photovoltaic water pumps for small communities in the semi-arid northeastern region in Brazil. **12th European Photovoltaic Solar Energy Conference**, 1994.

ARCH LINUX ARM. **Arch Linux ARM**. Disponível em: <<https://archlinuxarm.org/>>. Acesso em: 16 mar. 2016.

ARIAN CONTROL & INSTRUMENTACIÓN. **Pt100, su operación, instalación y tablas**, 2016. Disponível em: <<http://www.arian.cl>>. Acesso em: 3 jan. 2017

BARRETO, G. P. **Sistema de monitoramento e controle de consumo de energia elétrica para uma residência**. [s.l.] Centro universitário de Brasília - UNICEUB, 2015.

BENGHANEM, M.; ARAB, A. H.; MUKADAM, K. Data acquisition system for photovoltaic water pumps. **Renewable Energy**, v. 17, n. 3, p. 385–396, 1999.

BRITO, D. N. et al. Software Livre de Monitoramento Online de Microgeração Elétrica Via Modem WiFi. **II Encontro Nacional de Computação dos Institutos Federais**, p. 2–5, 2014.

CANALTECH. **O que é VNC? - Segurança.** Disponível em: <<http://corporate.canaltech.com.br/o-que-e/seguranca/o-que-e-vnc/>>. Acesso em: 29 mar. 2016.

CARVALHO, L. H. L. et al. Desenvolvimento de um sistema de monitoração para avaliação de desempenho operacional de um sistema fotovoltaico conectado à rede de baixa tensão. **VI Congresso Brasileiro de Energia Solar**, 2016.

CHILUNDO, R. J. **Sistema fotovoltaico aplicado à irrigação de uma unidade agrícola de produção familiar.** [s.l.] Universidade Federal do Ceará, 2014.

CRESESB. **Manual de Engenharia para Sistemas Fotovoltaicos.** [s.l.: s.n.].

CURL. **curl and libcurl.** Disponível em: <<https://curl.haxx.se/>>. Acesso em: 29 mar. 2016.

D-ROBOTICS UK. DHT11 Humidity & Temperature Sensor. **DHT11 Datasheet**, 2010.

DEVARAJAN, D. **A VHDL software model for networking smart transducers through Bluetooth technology.** [s.l.] Ohio State University, 2001.

DIAS, R. M.; DIAS, J. B.; PRÁ, L. B. D. Aquisição de dados de um sistema fotovoltaico conectado à rede com a plataforma Arduino. **VI Congresso Brasileiro de Energia Solar**, 2016.

ESTEVE, J. J.; BOLDRITO, R. S. **Administración avanzada de GNU-Linux : Software libre.** [s.l.: s.n.].

FATING, A. S.; NAGRARE, T. H. An Approach to Develop Embedded System For Web

Based Monitoring & Controlling of Renewable Energy Sources. **IJCSN International Journal of Computer Science and Network**, v. 2, n. 3, p. 86–90, 2013.

FEDRIZZI, M. C. **Sistemas fotovoltaicos de abastecimento de água para uso comunitário: Lições aprendidas e procedimentos para potencializar sua difusão**. [s.l.] Universidade de São Paulo, 2003.

FEDRIZZI, M. C.; SAUER, I. L. Bombeamento solar fotovoltaico, histórico, características e projetos. **4to Encontro de Energia no Meio Rural**, 2002.

GCC. **GCC online documentation**. Disponível em: <<https://gcc.gnu.org/onlinedocs/>>.

GNU. **O Projeto GNU**. Disponível em: <<http://www.gnu.org/gnu/thegnuproject.html>>. Acesso em: 30 mar. 2016.

GONZALES, D. A. **Prototipo de sistema de bombeo fotovoltaico para proyectos de cooperación al desarrollo con tecnologías apropiadas**. [s.l.] UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITÉCNICA SUPERIOR, 2010.

GRUPO ECIL. **Sensores PT 100**. Disponível em: <<http://www.ecil.com.br/temperatura-industrial/sensores-pt100/>>. Acesso em: 3 jan. 2017.

HALMEMAN, R. J. **Desenvolvimento de um sistema para monitoramento remoto em centrais de microgeração fotovoltaica**. [s.l.] Universidade Estadual Paulista “Julio De Mesquita Filho”, 2014.

HUKSEFLUX. LP02 Second class pyranometer. p. 1–2, 2013.

IEEE. **IEEE 802.11 Wireless Local Area Networks**. Disponível em: <<http://www.ieee802.org/11/>>. Acesso em: 6 out. 2015.

IFM ELECTRONIC. **Operating Instructions Ultrasonic Testing Instrument SU7000**, 2010a. Disponível em: <[www.ifm.com](http://www.ifm.com)>

IFM ELECTRONIC. Sensores de pressão: PN2094. p. 10–13, 2010b.

IOVINE, J. **PIC microcontroller project book**. United States of America: McGraw-Hil, 2000.

JUCÁ, S. C. S. et al. Sistema de Supervisão e Aquisição de Dados Aplicado a Bombeamento Fotovoltaico. **II Congresso Brasileiro de Energia Solar - CBENS**, n. 2, 2008.

JUCÁ, S. C. S. **Sistema de aquisição sem fio, supervisão via web e transmissão GPRS aplicados à microgeração descentralizada utilizando fontes renováveis de energia**. [s.l.] Universidade Federal Do Ceará, 2014.

JUCÁ, S. C. S.; CARVALHO, P. C. M.; BRITO, F. T. SanUSB: software educacional para o ensino da tecnologia de microcontroladores. **Ciências & Cognição**, v. 14, n. 3, p. 134–144, 2009.

JUCÁ, S. C. S.; CARVALHO, P. C. M.; BRITO, F. T. A low cost concept for data acquisition systems applied to decentralized renewable energy plants. **Sensors**, v. 11, n. 1, p. 743–756, 2011.

KALAITZAKIS, K.; KOUTROULIS, E.; VLACHOS, V. Development of a data acquisition system for remote monitoring of renewable energy systems. **Measurement**, v. 34, p. 75–83, 2003.

KEITHLEY INSTRUMENTS. **Data Acquisition and Control Handbook: A guide to hardware and software for computer-based measurement and control**. U.S.A.: Keithley Instruments, Inc., 2001.

KEPES, B. Understanding the Cloud Computing Stack SaaS, Paas, IaaS. **Rackspace support network**, p. 1–20, 2013.

KRISHNA, A. M. et al. Data Acquisition System for Performance Monitoring of Solar Photovoltaic ( PV ) Power Generation. **IJERT nternational Journal of Engineering Research & Technology**, v. 1, n. 7, p. 1–6, 2012.

KYOCERA. Solar Water Pumping Applications Guide. **Water**, 2002.

LABRADA, G. A.; HERNÁNDEZ, J. A. C. **Desarrollo de un sistema de adquisición de datos con microcontrolador PIC24 y comunicación inalámbrica WiFi**. [s.l.] Instituto Superior Politécnico - José Antonio Echeverría, 2007.

LABRADOR, R. M. G. **Administración De Servidores Linux (Ubuntu/Fedora )**. [s.l: s.n.].

MAHJOUBI, A.; MECHLOUCH, R. F.; BRAHIM, A. BEN. A low cost wireless data acquisition system for a remote photovoltaic (PV) water pumping system. **Energies**, v. 4, n. 1, p. 68–89, 2011.

MANITO, A. R. A. et al. Desenvolvimento de um sistema de aquisição de dados para monitoração de um sistema fotovoltaico conectado à rede. **III Congresso Brasileiro de Energia Solar**, 2010.

MEASUREMENT COMPUTING. **Data Acquisition Handbook: A Reference For DAQ And Analog & Digital Signal Conditioning**. Third Edit ed. United States of America: Measurement Computing Corporation, 2012.

MELL, P.; GRANCE, T. The NIST Definition of Cloud Computing. 2011.

MENDONÇA, D. E. **Planta de osmose reversa acionada por módulos fotovoltaicos com MPPT e válvula autoreguladora de pressão**. [s.l.] Universidade Federal do Ceará, 2016.

MENEZES, T. P. DE. **Sistema remoto de aquisição e monitoramento de variáveis ambientais para aplicações de previsão meteorológica**. [s.l.] Universidade Federal do Ceará, 2015.

MICROCHIP. PIC18F2455/2550/4455/4550 Data Sheet. **Technology**, 2006.

MOHANTY, P.; MUNEER, T.; KOLHE, M. **Solar Photovoltaic System Applications: A Guidebook for Off-Grid Electrification**. [s.l: s.n.].

MYSQL. **MySQL 5.7 Reference Manual 1 General Information**. Disponível em: <<http://dev.mysql.com/doc/refman/5.7/en/introduction.html>>. Acesso em: 11 abr. 2016.

NETO, A. C. DE Q.; BRITO, F. T.; JUCÁ, S. C. S. Sistema microcontrolado de baixo custo para aquisição, transmissão e supervisão de plantas de energias alternativas. **V Congresso Norte Nordeste de Pesquisa e Inovação**, 2010.

NOVUS. Transmissor TxRail 4-20 mA. n. 51, p. 19–21, 2016.

OMEGA ENGINEERING BRASIL. **Sensores Pt-100**. Disponível em: <<http://br.omega.com/prodinfo/pt100.html>>. Acesso em: 3 jan. 2017.

OPENELEC. **OpenELEC Mediacenter - Home**. Disponível em: <<http://openelec.tv/>>. Acesso em: 16 mar. 2016.

OPENWRT. **OpenWrt for Raspberry Pi**. Disponível em: <<https://openwrt.org/>>. Acesso em: 16 mar. 2016.

PEREIRA, R. I. S. **Projeto e construção de sistema embarcado para monitoramento online via WiFi aplicado à microgeração de módulos termoeletrônicos**. [s.l.] Universidade Federal Do Ceará, 2014.

PHP. **PHP: Prefácio - Manual**. Disponível em: <[http://php.net/manual/pt\\_BR/preface.php](http://php.net/manual/pt_BR/preface.php)>. Acesso em: 11 abr. 2016a.

PHP. **PHP: ¿Qué puede hacer PHP? - Manual**. Disponível em: <<http://php.net/manual/es/intro-whatcando.php>>. Acesso em: 11 abr. 2016b.

PIDORA. **Pidora - Raspberry Pi Fedora Remix**. Disponível em: <<http://pidora.ca/>>. Acesso em: 16 mar. 2016.

RASPBERRY DIGITAL SIGNAGE. **Raspberry Digital Signage | Binary Emotions**. Disponível em: <[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/Deployment\\_Guide/ch-autotasks.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/ch-autotasks.html)>. Acesso em:

16 mar. 2016.

RASPBERRY FOUNDATION. **Raspberry Pi 2 Model B**. Disponível em: <<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>>. Acesso em: 15 mar. 2016.

RED HAT ENTERPRISE LINUX 4. **Manual de referencia - Protocolo SSH**. Disponível em: <<http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html>>. Acesso em: 29 mar. 2016a.

RED HAT ENTERPRISE LINUX 4. **Manual de referencia - Samba**. Disponível em: <<http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-samba.html>>. Acesso em: 29 mar. 2016b.

RED HAT ENTERPRISE LINUX 5. **Deployment Guide - Automated Tasks**. Disponível em: <[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/Deployment\\_Guide/ch-autotasks.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/ch-autotasks.html)>. Acesso em: 4 abr. 2016.

RISC OS. **RISC OS Open: Welcome**. Disponível em: <<https://www.riscosopen.org/content/>>. Acesso em: 16 mar. 2016.

ROQUE, F. P.; ZALDÍVAR, E. V.; DE FUENTES, O. A. Sistema de adquisición de datos con comunicación inalámbrica. **RIELAC Revista de Ingeniería Electrónica, Automática y Comunicaciones**, v. 34, n. 3, p. 63–73, 2013.

SAMBA. **Samba - opening windows to a wider world**. Disponível em: <<https://www.samba.org/>>. Acesso em: 29 mar. 2016.

SHURFLO. PRODUCT DATA SHEET PUMP MODEL 8000-443-136. v. 25, n. 57, p. 1–2, 2008.

SILVA, G. M. DA. **Guia Foca GNU/Linux**. Disponível em: <<http://www.guiafoca.org/cgs/guia/avancado/index.html#contents>>. Acesso em: 11 abr. 2016.

SOUSA, M. L. DE A. **Uso de bombeamento fotovoltaico para irrigação de unidades de produção agrícola de pequeno porte.** [s.l.] Universidade Federal do Ceará, 2014.

SRCSL. **¿ Qué es un sensor PT100 ? | Sistemas de Regulación y Control, S.L.:: Especialistas en temperatura y control de procesos.** Disponível em: <<http://srcsl.com/que-es-un-sensor-pt100/>>. Acesso em: 3 jan. 2017.

TOLEDO ALMA, E. et al. MySQL. p. 1–2, 2016.

VMWARE. **Operating a VMware vCloud.** Disponível em: <<https://www.vmware.com/files/pdf/vcat/Operating-VMware-vCloud.pdf>>.

WIRINGPI. **WiringPi.** Disponível em: <<http://wiringpi.com/>>. Acesso em: 30 mar. 2016.

YINGLI SOLAR. **Yge 95 Series,** 2008. Disponível em: <[www.yinglisolar.com](http://www.yinglisolar.com)>

## APÊNDICES

### Apêndice A - Artigos completos publicados vinculados ao projeto de pesquisa

- Em congressos:

[1] DUPONT, I. M., PEREIRA, R. I. S., CARVALHO, P. C. M., JUCÁ, S. C. S. **Sistema embarcado Linux aplicado ao monitoramento em nuvem de planta de microgeração fotovoltaica.** VI Congresso Brasileiro de Energia Solar (CBENS), Belo Horizonte, 2016.

[2] ROCHA, J. I. C. F., DUPONT, I. M., CARNEIRO, A. V., CHAVES, C.M.N.M., FREITAS, D. M., HOLANDA, I. S., KERBAGE, J. C., ALVES, O. M., CARVALHO, P. C. M. **Estimativa do ganho proporcionado com o uso de seguimento solar para Fortaleza/CE.** VI Congresso Brasileiro de Energia Solar (CBENS), Belo Horizonte, 2016.

[3] CARNEIRO, A. V., DUPONT, I. M., Luna, M. L. M., HOLANDA, I. S., FREITAS, D. M., ALVES, O. M., KERBAGE, J. C., CHAVES, C.M.N.M., CARVALHO, P. C. M. **Utilização dos modelos de 4 e 5 parâmetros para módulos fotovoltaicos considerando condições climáticas próximas à linha do Equador.** VI Congresso Brasileiro de Energia Solar (CBENS), Belo Horizonte, 2016.

## Apêndice B - Catálogo técnico do Painei FV

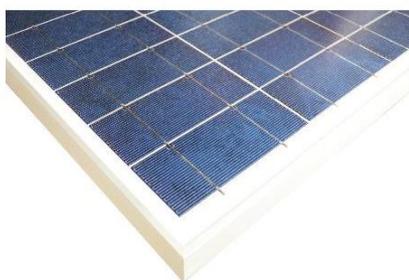
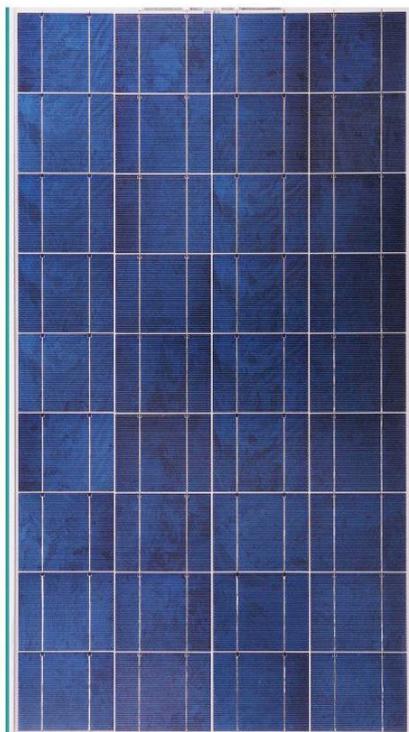
# YGE 95 SERIES

Powered by **YINGLI**

YL095P-17b 2/3

YL090P-17b 2/3

YL085P-17b 2/3



### ABOUT YINGLI GREEN ENERGY

Yingli Green Energy Holding Company Limited (NYSE: YGE) is one of the world's largest fully vertically integrated PV manufacturers. With over 4.5 GW of modules installed globally, we are a leading solar energy company built upon proven product reliability and sustainable performance. We are the first renewable energy company and the first Chinese company to sponsor the FIFA World Cup™.

### PERFORMANCE

High efficiency, polycrystalline solar cells with high transmission and textured glass delivering a module efficiency of up to 14.3%, minimizing installation costs and maximizing the kWh output of your system per unit area.

### QUALITY AND RELIABILITY

- Industry leading in-house manufacturing of polysilicon, ingots, wafers, cells and modules ensures tight control of our material and production quality.
- Robust, corrosion resistant aluminum frame independently tested to withstand wind loads of 2.4 kPa and snow loads of 2.4 kPa ensuring a stable mechanical life for your modules.
- Module packaging optimized to protect product during transportation and minimize on-site waste.
- This type of module is commonly use for the small off-grid system.

### QUALIFICATIONS AND CERTIFICATES

IEC 61215, IEC 61730, CGC, ISO 9001:2008



[YINGLISOLAR.COM](http://YINGLISOLAR.COM)



# YGE 95 SERIES

## ELECTRICAL PERFORMANCE

### Electrical parameters at Standard Test Conditions (STC)

Module name			YGE 95	YGE 90	YGE 85
Module type			YL95P-17b 2/3	YL90P-17b 2/3	YL85P-17b 2/3
Power output	$P_{max}$	W	95	90	85
Power output tolerances	$\Delta P_{max}$	%	+/- 5		
Module efficiency	$\eta_m$	%	14.3	13.5	12.7
Voltage at $P_{max}$	$V_{mp}$	V	18.18	17.77	17.40
Current at $P_{max}$	$I_{mp}$	A	5.23	5.06	4.89
Open-circuit voltage	$V_{oc}$	V	22.5	22.0	21.53
Short-circuit current	$I_{sc}$	A	5.59	5.44	5.19

STC: 1000W/m<sup>2</sup> irradiance, 25°C cell temperature, AM1.5g spectrum according to EN 60904-3.  
Average relative efficiency reduction of 5% at 200W/m<sup>2</sup> according to EN 60904-1.

### Electrical parameters at Nominal Operating Cell Temperature (NOCT)

Power output	$P_{max}$	W	76.96	72.91	68.86
Voltage at $P_{max}$	$V_{mp}$	V	16.75	16.38	16.03
Current at $P_{max}$	$I_{mp}$	A	4.20	4.06	3.92
Open-circuit voltage	$V_{oc}$	V	20.48	20.02	19.59
Short-circuit current	$I_{sc}$	A	4.47	4.35	4.15

NOCT: open-circuit module operation temperature at 800W/m<sup>2</sup> irradiance, 20°C ambient temperature, 1m/s wind speed.

## THERMAL CHARACTERISTICS

Nominal operating cell temperature	NOCT	°C	46 +/- 2
Temperature coefficient of $P_{max}$	$\gamma$	%/°C	-0.45
Temperature coefficient of $V_{oc}$	$\beta_{voc}$	%/°C	-0.33
Temperature coefficient of $I_{sc}$	$\alpha_{sc}$	%/°C	0.06

## OPERATING CONDITIONS

Max. system voltage	50V <sub>oc</sub>
Max. series fuse rating	10A
Limiting reverse current	Do not apply external voltages larger than Voc of the module
Operating temperature range	-40 to 85°C
Max. static load, front (e.g., snow and wind)	2400Pa
Max. static load, back (e.g., wind)	2400Pa
Max. hailstone impact (diameter / velocity)	25mm / 23m/s

## CONSTRUCTION MATERIALS

Front cover (material / thickness)	low-iron tempered glass / 3.2mm
Cell (quantity / material / dimensions)	36 / multicrystalline silicon / 156mm x 104mm
Encapsulant (material)	ethylene vinyl acetate (EVA)
Frame (material / color / anodization color)	anodized aluminum alloy / silver / clear

- Due to continuous innovation, research and product improvement, the specifications in this product information sheet are subject to change without prior notice. The specifications may deviate slightly and are not guaranteed.
- The data do not refer to a single module and they are not part of the offer, they only serve for comparison to different module types.

Yingli Green Energy Holding Co. Ltd.

service@yinglisolar.com

Tel: 0086-312-8929802

YINGLISOLAR.COM

© Yingli Green Energy Holding Co. Ltd. | DS\_YL095P-17b\_23\_EU\_EN\_201302\_v02.3

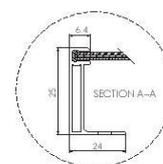
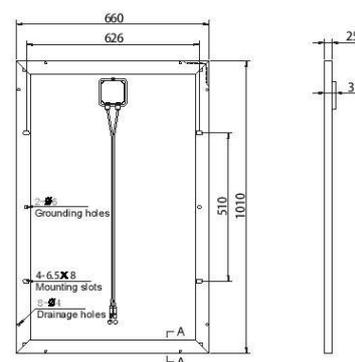
## GENERAL CHARACTERISTICS

Dimensions (L / W / H)	1010mm / 660mm / 25mm
Weight	7.65kg

## PACKAGING SPECIFICATIONS

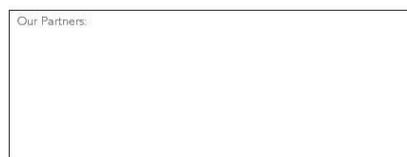
Number of modules per box	6
Packaging box dimensions (L / W / H)	1040mm / 690mm / 210mm

Unit: mm



Warning: Read the Installation and User Manual in its entirety before handling, installing, and operating Yingli Solar modules.

Our Partners:



## Apêndice C - Catálogo técnico do conjunto motobomba

# SHURflo®

First in Fluid Innovation

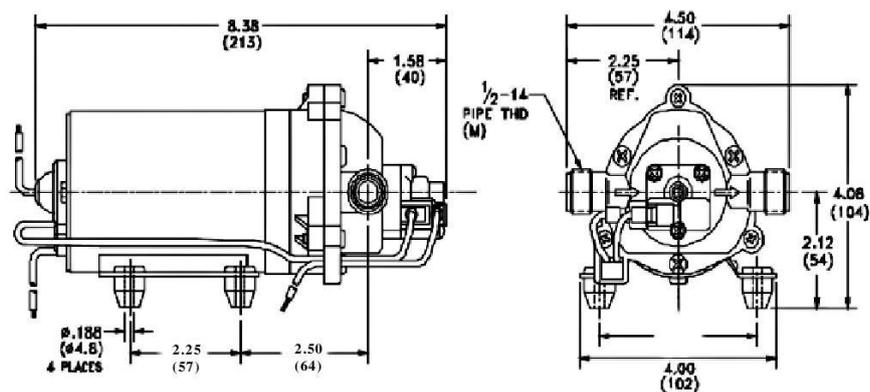
## PRODUCT DATA SHEET

MODEL: 8000-443-136

### SPECIFICATIONS:

**MODEL NUMBER:** 8000-443-136  
**PUMP DESIGN:** Positive Displacement 3 Chamber Diaphragm Pump  
**CHECK VALVE:** (1-Way Operation) Prevents Reverse Flow  
**CAM:** 3.0 Degree  
**MOTOR:** Permanent Magnet, P/N 11-111-00  
**VOLTAGE:** 12 VDC Nominal  
**PRESSURE SWITCH:** Adjustable Shut-Off (Range 40-60 PSI)  
 Factory Set @ 60 PSI, Turn On 45PSI  $\pm$  5 PSI  
**LIQUID TEMPERATURE:** 170 Degrees Fahrenheit (77 Degrees Centigrade) Max.  
**PRIME:** Self-Priming Up To 12 Ft. Vertical,  
 Max. Inlet Pressure 30 PSI (2.1 Bar)  
**PORTS:** 1/2"-14 Male Parallel Thread  
**MATERIAL OF CONSTRUCTION:**  
**PLASTICS-** Polypropylene  
**VALVES-** EPDM  
**DIAPHRAGM-** Santoprene  
**FASTENERS-** Zinc Plated Steel  
**NET WEIGHT:** 4.56 Lbs (2.07Kg)  
**DUTY CYCLE:** Continuous (See Temperature Rise Chart)  
**TYPICAL APPLICATIONS:** Water Delivery

### DIMENSIONS:



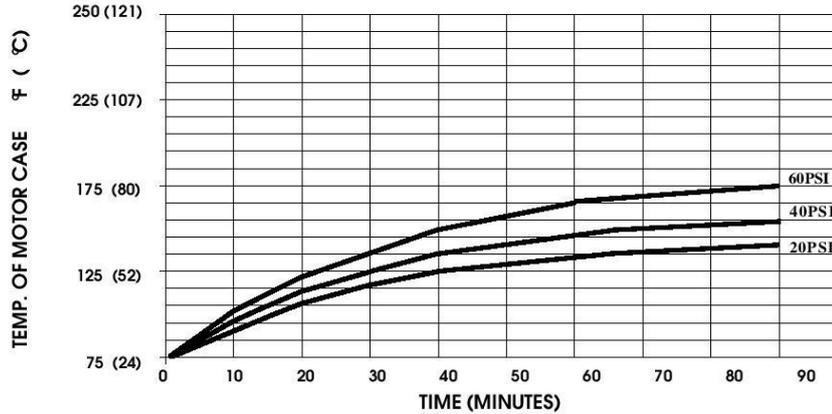
FILE #8000-443-136

REVISED:

ISSUED: 4/1/93

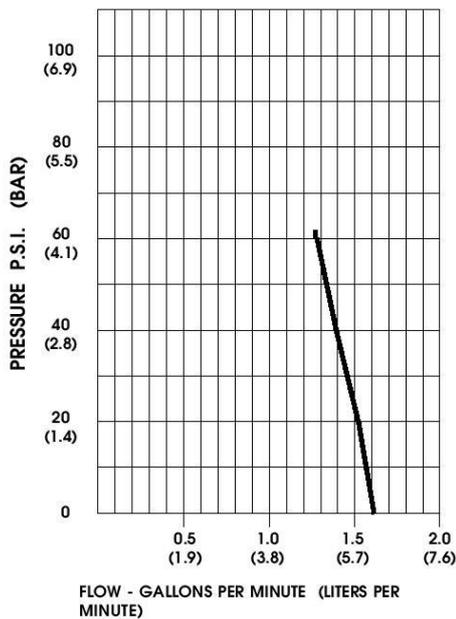
MODEL: 8000-443-136

**TEMPERATURE RISE**



THIS GRAPH IS FOR USE AS A DESIGN GUIDE. IT IS BASED ON RUNNING CONTINUOUSLY WITH AN AMBIENT TEMPERATURE OF 70° F IN STILL AIR.

**TYPICAL PERFORMANCE**



PRESSURE (PSI)	FLOW (GPM/LIT)	RPM MIN/MAX	CURRENT (AMPS)	VOLTAGE (VOLTS)
OPEN	1.75/6.6	2290/2315	3.1	12VDC
10	1.66/6.3	225/2255	3.4	"
20	1.57/5.9	2170/2205	4.2	"
30	1.48/5.6	2130/2155	4.9	"
40	1.38/5.2	2045/2085	5.6	"
50	1.30/4.9	2020/2045	6.9	"
60	1.23/4.6	1893/1970	7.2	"

*-SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.  
-ALL DATA BASED ON TESTING WITH WATER AT AMBIENT TEMPERATURE.*

## Apêndice D - Catálogo técnico do transmissor do sensor PT-100 e tabelas de correlação da resistência e temperatura

### NOVUS TERMORRESISTÊNCIAS Pt100

#### APRESENTAÇÃO

Termorresistências de platina são sensores de temperatura cujo princípio de funcionamento se baseia na alteração da resistência elétrica do elemento com a variação da temperatura.

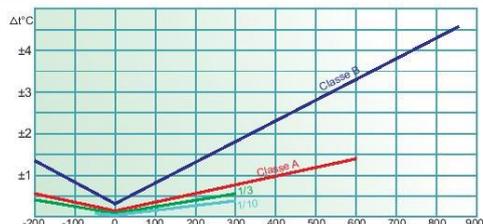
A variação da resistência com a temperatura é chamada de coeficiente de temperatura e é especificada como a média da variação entre 0 e 100 °C. Este coeficiente é de 0,385 por °C, conforme a norma IEC 60751.

Tanto nos modelos de fio bobinado quanto nos modelos de filme plano, onde uma película de platina é depositada por "sputtering" em alto vácuo sobre o substrato cerâmico, a estabilidade em altas temperaturas é uma constante.



#### TOLERÂNCIAS CONFORME A NORMA IEC 60751

Range °C	Ohms	Classe A		Classe B	
		Ohms	°C	Ohms	°C
-200	18,52	±0,24	±0,55	±0,56	±1,3
-100	60,26	±0,14	±0,35	±0,32	±0,8
0	100,00	±0,06	±0,15	±0,12	±0,3
100	138,51	±0,13	±0,35	±0,30	±0,8
200	175,86	±0,20	±0,55	±0,48	±1,3
300	212,05	±0,27	±0,75	±0,64	±1,8
400	247,09	±0,33	±0,95	±0,79	±2,3
500	280,98	±0,38	±1,15	±0,93	±2,8
600	313,71	±0,43	±1,35	±1,06	±3,3
700	345,28	-	-	±1,17	±3,8
800	375,70	-	-	±1,28	±4,3
850	390,48	-	-	±1,34	±4,6



#### SÉRIE CRZ



TAMANHO REAL	MODELO	DIMENSÕES (mm) D L A	OHMS a 0°C	CLASSE	CORRENTE DE MEDIDA	COMPRI- MENTO TERMINAIS	TEMP. DE OPERAÇÃO	RESPOSTA EM (s)
	CRZ-2005	2 x 5 x 1,5	100	A, B	2 mA máx.	12 ± 3 mm	-50 a 600°C	0,3*

\* Tempo em segundos para 90% da resposta de 0°C a 100°C em água agitada

#### SÉRIE CR

- São confeccionadas em corpo cerâmico de alta alumina contendo capilares onde é inserido o filamento de platina.
- Os capilares são preenchidos com finíssimo pó cerâmico mantendo o filamento livre de vibrações sem causar "stress" mecânico em altas temperaturas.
- Com aplicações entre -200°C e 650°C, permitem uma resposta livre de histerese.
- Disponíveis nas classes de precisão A e B, conforme a norma IEC 60751, podem ser adquiridas nas classes 1/3 e 1/10 de DIN sob consulta.



TAMANHO REAL	MODELO	DIMENSÕES (mm) D L	CORRENTE DE MEDIDA	COMPRI- MENTO TERMINAIS	CLASSE	TEMP. DE OPERAÇÃO
	CR-1010	1 ± 0,1 x 10 ± 0,2	1 mA máx.	10 ± 3 mm	A, B	-200 a 650°C
	CR-1515	1,5 ± 0,1 x 15 ± 0,2	2 mA máx.	10 ± 3 mm	A, B	-200 a 650°C
	CR-2020	2,0 ± 0,1 x 20 ± 0,2	5 mA máx.	10 ± 3 mm	A, B	-200 a 650°C
	CR-3030	3,0 ± 0,1 x 30 ± 0,2	5 mA máx.	13 ± 3 mm	A, B	-200 a 650°C

NOVUS

## Pt100 - TABELA DA RESISTÊNCIA COM A TEMPERATURA

T (°C)	R (Ω)	R (Ω) - T + 1°C	R (Ω) - T + 2°C	R (Ω) - T + 3°C	R (Ω) - T + 4°C	R (Ω) - T + 5°C	R (Ω) - T + 6°C	R (Ω) - T + 7°C	R (Ω) - T + 8°C	R (Ω) - T + 9°C	Tolerância em °C	
											Cl. A	Cl. B
-200	18,49	18,93	19,36	19,79	20,22	20,65	21,08	21,51	21,94	22,37	± 0,55	± 1,30
-190	22,80	23,23	23,66	24,09	24,52	24,94	25,37	25,80	26,23	26,65	± 0,53	± 1,25
-180	27,08	27,50	27,93	28,35	28,78	29,20	29,63	30,05	30,47	30,90	± 0,51	± 1,20
-170	31,32	31,74	32,16	32,59	33,01	33,43	33,85	34,27	34,69	35,11	± 0,49	± 1,15
-160	35,53	35,95	36,37	36,79	37,21	37,63	38,04	38,46	38,88	39,30	± 0,47	± 1,10
-150	39,71	40,13	40,55	40,96	41,38	41,79	42,21	42,63	43,04	43,45	± 0,45	± 1,05
-140	43,87	44,28	44,70	45,11	45,52	45,94	46,35	46,76	47,18	47,59	± 0,43	± 1,00
-130	48,00	48,41	48,82	49,23	49,64	50,06	50,47	50,88	51,29	51,70	± 0,41	± 0,95
-120	52,11	52,52	52,92	53,33	53,74	54,15	54,56	54,97	55,38	55,78	± 0,39	± 0,90
-110	56,19	56,60	57,00	57,41	57,82	58,22	58,63	59,04	59,44	59,85	± 0,37	± 0,85
-100	60,25	60,66	61,06	61,47	61,87	62,28	62,68	63,09	63,49	63,90	± 0,35	± 0,80
-90	64,30	64,70	65,11	65,51	65,91	66,31	66,72	67,12	67,52	67,92	± 0,33	± 0,75
-80	68,33	68,73	69,13	69,53	69,93	70,33	70,73	71,13	71,53	71,93	± 0,31	± 0,70
-70	72,33	72,73	73,13	73,53	73,93	74,33	74,73	75,13	75,53	75,93	± 0,29	± 0,65
-60	76,33	76,73	77,13	77,52	77,92	78,32	78,72	79,11	79,51	79,91	± 0,27	± 0,60
-50	80,31	80,70	81,10	81,50	81,89	82,29	82,69	83,08	83,48	83,88	± 0,25	± 0,55
-40	84,27	84,67	85,06	85,46	85,85	86,25	86,64	87,04	87,43	87,83	± 0,23	± 0,50
-30	88,22	88,62	89,01	89,40	89,80	90,19	90,59	90,98	91,37	91,77	± 0,21	± 0,45
-20	92,16	92,55	92,95	93,34	93,73	94,12	94,52	94,91	95,30	95,69	± 0,19	± 0,40
-10	96,09	96,48	96,87	97,26	97,65	98,04	98,44	98,83	99,22	99,61	± 0,17	± 0,35
0	100,00	100,39	100,78	101,17	101,56	101,95	102,34	102,73	103,12	103,51	± 0,15	± 0,30
10	103,90	104,29	104,68	105,07	105,46	105,85	106,24	106,63	107,02	107,40	± 0,17	± 0,35
20	107,79	108,18	108,57	108,96	109,35	109,73	110,12	110,51	110,90	111,28	± 0,19	± 0,40
30	111,67	112,06	112,45	112,83	113,22	113,61	113,99	114,38	114,77	115,15	± 0,21	± 0,45
40	115,54	115,93	116,31	116,70	117,08	117,47	117,85	118,24	118,62	119,01	± 0,23	± 0,50
50	119,40	119,78	120,16	120,55	120,93	121,32	121,70	122,09	122,47	122,86	± 0,25	± 0,55
60	123,24	123,62	124,01	124,39	124,77	125,16	125,54	125,92	126,31	126,69	± 0,27	± 0,60
70	127,07	127,45	127,84	128,22	128,60	128,98	129,37	129,75	130,13	130,51	± 0,29	± 0,65
80	130,89	131,27	131,66	132,04	132,42	132,80	133,18	133,56	133,94	134,32	± 0,31	± 0,70
90	134,70	135,08	135,46	135,84	136,22	136,60	136,98	137,36	137,74	138,12	± 0,33	± 0,75
100	138,50	138,88	139,26	139,64	140,02	140,39	140,77	141,15	141,53	141,91	± 0,35	± 0,80
110	142,29	142,66	143,04	143,42	143,80	144,17	144,55	144,93	145,31	145,68	± 0,37	± 0,85
120	146,06	146,44	146,81	147,19	147,57	147,94	148,32	148,70	149,07	149,45	± 0,39	± 0,90
130	149,82	150,20	150,57	150,95	151,33	151,70	152,08	152,45	152,83	153,20	± 0,41	± 0,95
140	153,58	153,95	154,32	154,70	155,07	155,45	155,82	156,19	156,57	156,94	± 0,43	± 1,00
150	157,31	157,69	158,06	158,43	158,81	159,18	159,55	159,93	160,30	160,67	± 0,45	± 1,05
160	161,04	161,42	161,79	162,16	162,53	162,90	163,27	163,65	164,02	164,39	± 0,47	± 1,10
170	164,76	165,13	165,50	165,87	166,24	166,61	166,98	167,35	167,72	168,09	± 0,49	± 1,15
180	168,46	168,83	169,20	169,57	169,94	170,31	170,68	171,05	171,42	171,79	± 0,51	± 1,20
190	172,16	172,53	172,90	173,26	173,63	174,00	174,37	174,74	175,10	175,47	± 0,53	± 1,25
200	175,84	176,21	176,57	176,94	177,31	177,68	178,04	178,41	178,78	179,14	± 0,55	± 1,30
210	179,51	179,88	180,24	180,61	180,97	181,34	181,71	182,07	182,44	182,80	± 0,57	± 1,35
220	183,17	183,53	183,90	184,26	184,63	184,99	185,36	185,72	186,09	186,45	± 0,59	± 1,40
230	186,82	187,18	187,54	187,91	188,27	188,63	189,00	189,36	189,72	190,09	± 0,61	± 1,45
240	190,45	190,81	191,18	191,54	191,90	192,26	192,63	192,99	193,35	193,71	± 0,63	± 1,50
250	194,07	194,44	194,80	195,16	195,52	195,88	196,24	196,60	196,96	197,33	± 0,65	± 1,55
260	197,69	198,05	198,41	198,77	199,13	199,49	199,85	200,21	200,57	200,93	± 0,67	± 1,60
270	201,29	201,65	202,01	202,36	202,72	203,08	203,44	203,80	204,16	204,52	± 0,69	± 1,65
280	204,88	205,23	205,59	205,95	206,31	206,67	207,02	207,38	207,74	208,10	± 0,71	± 1,70
290	208,45	208,81	209,17	209,52	209,88	210,24	210,59	210,95	211,31	211,66	± 0,73	± 1,75
300	212,02	212,37	212,73	213,09	213,44	213,80	214,15	214,51	214,86	215,22	± 0,75	± 1,80
310	215,57	215,93	216,28	216,64	216,99	217,35	217,70	218,05	218,41	218,76	± 0,77	± 1,85
320	219,12	219,47	219,82	220,18	220,53	220,88	221,24	221,59	221,94	222,29	± 0,79	± 1,90
330	222,65	223,00	223,35	223,70	224,06	224,41	224,76	225,11	225,46	225,81	± 0,81	± 1,95
340	226,17	226,52	226,87	227,22	227,57	227,92	228,27	228,62	228,97	229,32	± 0,83	± 2,00
350	229,67	230,02	230,37	230,72	231,07	231,42	231,77	232,12	232,47	232,82	± 0,85	± 2,05
360	233,17	233,52	233,87	234,22	234,56	234,91	235,26	235,61	235,96	236,31	± 0,87	± 2,10
370	236,65	237,00	237,35	237,70	238,04	238,39	238,74	239,09	239,43	239,78	± 0,89	± 2,15
380	240,13	240,47	240,82	241,17	241,51	241,86	242,20	242,55	242,90	243,24	± 0,91	± 2,20
390	243,59	243,93	244,28	244,62	244,97	245,31	245,66	246,00	246,35	246,69	± 0,93	± 2,25
400	247,04	247,38	247,73	248,07	248,41	248,76	249,10	249,45	249,79	250,13	± 0,95	± 2,30

SENSORES DE TEMPERATURA E ACESSÓRIOS



G6

info@novus.com.br  
www.novus.com.br

NOVUS PRODUTOS ELETRÔNICOS LTDA  
TEL. 55 51 3323-3600 FAX: 55 51 3323-3645

NOVUS

## Pt100 - TABELA DA RESISTÊNCIA COM A TEMPERATURA

T (°C)	R (Ω)	R (Ω) - T + 1°C	R (Ω) - T + 2°C	R (Ω) - T + 3°C	R (Ω) - T + 4°C	R (Ω) - T + 5°C	R (Ω) - T + 6°C	R (Ω) - T + 7°C	R (Ω) - T + 8°C	R (Ω) - T + 9°C	Tolerância em °C	
											Cl. A	Cl. B
410	250,48	250,82	251,16	251,50	251,85	252,19	252,53	252,88	253,22	253,56	± 0,97	± 2,35
420	253,90	254,24	254,59	254,93	255,27	255,61	255,95	256,29	256,64	256,98	± 0,99	± 2,40
430	257,32	257,66	258,00	258,34	258,68	259,02	259,36	259,70	260,04	260,38	± 1,01	± 2,45
440	260,72	261,06	261,40	261,74	262,08	262,42	262,76	263,10	263,43	263,77	± 1,03	± 2,50
450	264,11	264,45	264,79	265,13	265,47	265,80	266,14	266,48	266,82	267,15	± 1,05	± 2,55
460	267,49	267,83	268,17	268,50	268,84	269,18	269,51	269,85	270,19	270,52	± 1,07	± 2,60
470	270,86	271,20	271,53	271,87	272,20	272,54	272,88	273,21	273,55	273,88	± 1,09	± 2,65
480	274,22	274,55	274,89	275,22	275,56	275,89	276,23	276,56	276,89	277,23	± 1,11	± 2,70
490	277,56	277,90	278,23	278,56	278,90	279,23	279,56	279,90	280,23	280,56	± 1,13	± 2,75
<b>500</b>	<b>280,90</b>	<b>281,23</b>	<b>281,56</b>	<b>281,89</b>	<b>282,23</b>	<b>282,56</b>	<b>282,89</b>	<b>283,22</b>	<b>283,55</b>	<b>283,89</b>	<b>± 1,15</b>	<b>± 2,80</b>
510	284,22	284,55	284,88	285,21	285,54	285,87	286,21	286,54	286,87	287,20	± 1,17	± 2,85
520	287,53	287,86	288,19	288,52	288,85	289,18	289,51	289,84	290,17	290,50	± 1,19	± 2,90
530	290,83	291,16	291,49	291,81	292,14	292,47	292,80	293,13	293,46	293,79	± 1,21	± 2,95
540	294,11	294,44	294,77	295,10	295,43	295,75	296,08	296,41	296,74	297,07	± 1,23	± 3,00
550	297,39	297,72	298,04	298,37	298,70	299,02	299,35	299,68	300,00	300,33	± 1,25	± 3,05
560	300,65	300,98	301,31	301,63	301,96	302,28	302,61	302,93	303,26	303,58	± 1,27	± 3,10
570	303,91	304,23	304,56	304,88	305,20	305,53	305,85	306,18	306,50	306,82	± 1,29	± 3,15
580	307,15	307,47	307,79	308,12	308,44	308,76	309,09	309,41	309,73	310,05	± 1,31	± 3,20
590	310,38	310,70	311,02	311,34	311,67	311,99	312,31	312,63	312,95	313,27	± 1,33	± 3,25
<b>600</b>	<b>313,59</b>	<b>313,92</b>	<b>314,24</b>	<b>314,56</b>	<b>314,88</b>	<b>315,20</b>	<b>315,52</b>	<b>315,84</b>	<b>316,16</b>	<b>316,48</b>	<b>± 1,35</b>	<b>± 3,30</b>
610	316,80	317,12	317,44	317,76	318,08	318,40	318,72	319,04	319,36	319,68	± 1,37	± 3,35
620	319,99	320,31	320,63	320,95	321,27	321,59	321,91	322,22	322,54	322,86	± 1,39	± 3,40
630	323,18	323,49	323,81	324,13	324,45	324,76	325,08	325,40	325,72	326,03	± 1,41	± 3,45
640	326,35	326,66	326,98	327,30	327,61	327,93	328,25	328,56	328,88	329,19	± 1,43	± 3,50
650	329,51	329,82	330,14	330,45	330,77	331,08	331,40	331,71	332,03	332,34	± 1,45	± 3,55
660	332,66	332,97	333,28	333,60	333,91	334,23	334,54	334,85	335,17	335,48		± 3,60
670	335,79	336,11	336,42	336,73	337,04	337,36	337,67	337,98	338,29	338,61		± 3,65
680	338,92	339,23	339,54	339,85	340,16	340,48	340,79	341,10	341,41	341,72		± 3,70
690	342,03	342,34	342,65	342,96	343,27	343,58	343,89	344,20	344,51	344,82		± 3,75
<b>700</b>	<b>345,13</b>	<b>345,44</b>	<b>345,75</b>	<b>346,06</b>	<b>346,37</b>	<b>346,68</b>	<b>346,99</b>	<b>347,30</b>	<b>347,60</b>	<b>347,91</b>		<b>± 3,80</b>
710	348,22	348,53	348,84	349,15	349,45	349,76	350,07	350,38	350,69	350,99		± 3,85
720	351,30	351,61	351,91	352,22	352,53	352,83	353,14	353,45	353,75	354,06		± 3,90
730	354,37	354,67	354,98	355,28	355,59	355,90	356,20	356,51	356,81	357,12		± 3,95
740	357,42	357,73	358,03	358,34	358,64	358,95	359,25	359,55	359,86	360,16		± 4,00
750	360,47	360,77	361,07	361,38	361,68	361,98	362,29	362,59	362,89	363,19		± 4,05
760	363,50	363,80	364,10	364,40	364,71	365,01	365,31	365,61	365,91	366,22		± 4,10
770	366,52	366,82	367,12	367,42	367,72	368,02	368,32	368,63	368,93	369,23		± 4,15
780	369,53	369,83	370,13	370,43	370,73	371,03	371,33	371,63	371,93	372,22		± 4,20
790	372,52	372,82	373,12	373,42	373,72	374,02	374,32	374,61	374,91	375,21		± 4,25
<b>800</b>	<b>375,51</b>	<b>375,81</b>	<b>376,10</b>	<b>376,40</b>	<b>376,70</b>	<b>377,00</b>	<b>377,29</b>	<b>377,59</b>	<b>377,89</b>	<b>378,19</b>		<b>± 4,30</b>
810	378,48	378,78	379,08	379,37	379,67	379,97	380,26	380,56	380,85	381,15		± 4,35
820	381,45	381,74	382,04	382,33	382,63	382,92	383,22	383,51	383,81	384,10		± 4,40
830	384,40	384,69	384,98	385,28	385,57	385,87	386,16	386,45	386,75	387,04		± 4,45
840	387,34	387,63	387,92	388,21	388,51	388,80	389,09	389,39	389,68	389,97		± 4,50
850	390,26	390,55	390,85	391,14	391,43	391,72	392,01	392,30	392,60	392,89		± 4,55

SENSORES DE TEMPERATURA E ACESSÓRIOS

## Fórmulas para conversão (conforme DIN IEC 751)

Para o range -200...0 °C:

$$R[\Omega] = Pt \left[ 1 + 3,90802 \cdot 10^{-3} \cdot t - 0,580195 \cdot 10^{-6} \cdot t^2 - 4,27350 \cdot 10^{-12} \cdot (t - 100) \cdot t^3 \right]$$

Para o range 0...850 °C:

$$R[\Omega] = Pt \left( 1 + 3,90802 \cdot 10^{-3} \cdot t - 0,580195 \cdot 10^{-6} \cdot t^2 \right)$$

Onde:

R = resistência em Ohms

Pt = valor Ôhmico a zero graus ("100")

t = temperatura

## Apêndice E – Script para execução do Ustream no Rpi

```
#!/bin/bash

myVideo=`ls /dev |grep video`

case "$1" in
  start)
    echo "Starting ustream"
    avconv -f video4linux2 -r 10 -b 350k -i /dev/$myVideo -f flv
    rtmp://1.22232789.fme.ustream.tv/ustreamVideo/22232789/4HnSLGB2jP2bxyVfUZDGQnZfuqc5fLJ7 &
    ;;
  stop)
    echo "Stopping ustream"
    killall avconv
    ;;
  *)
    echo "Usage: ustream [start|stop]"
    exit 1
    ;;
esac

exit 0
```

## Apêndice F – Script para obter temperatura e umidade do DHT1

```

#include <wiringPi.h>
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>

//Definimos constantes
#define MAX_TIME 85
#define DHT11PIN 7
#define ATTEMPTS 5

//Definimos un vector global
int dht11_val[5]={0,0,0,0,0};

////////////////////////////////////
//Funcion principal para leer los valores del sensor.
int dht11_read_val(){
    uint8_t lststate=HIGH;
    uint8_t counter=0;
    uint8_t j=0,i;
    for(i=0;i<5;i++){
        dht11_val[i]=0;
    }
    pinMode(DHT11PIN,OUTPUT);
    digitalWrite(DHT11PIN,LOW);
    delay(18);
    digitalWrite(DHT11PIN,HIGH);
    delayMicroseconds(40);
    pinMode(DHT11PIN,INPUT);
    for(i=0;i<MAX_TIME;i++){
        counter=0;
        while(digitalRead(DHT11PIN)==lststate){
            counter++;
            delayMicroseconds(1);
            if(counter==255){
                break;
            }
        }
    }
}

```

```

}
lststate=digitalRead(DHT11PIN);
if(counter==255){
  break;
}
//Las 3 primeras transiciones son ignoradas
if((i>=4)&&(i%2==0)){
  dht11_val[j/8]<=1;
  if(counter>16){
    dht11_val[j/8]=1;
    //printf("%d\n",dht11_val[j/8]);
  }
  j++;
}
}

char str[100] = "echo ";
char strh[10];
char strh1[10];
char strt[10];
char strt1[10];

// Hacemos una suma de comprobacion para ver si el dato es correcto. Si es asi, lo mostramos
if((j>=40)&&(dht11_val[4]==((dht11_val[0]+dht11_val[1]+dht11_val[2]+dht11_val[3])& 0xFF)))
{
  printf("%d.%d,%d.%d\n",dht11_val[0],dht11_val[1],dht11_val[2],dht11_val[3]);
}

else {
  return 0;
}
}

////////////////////////////////////
//Empieza nuestro programa principal.
int main(void){
  //Establecemos el numero de intentos que vamos a realizar
  //la constante ATTEMPTS esta definida arriba
  int attempts=ATTEMPTS;

```

```
//Si la libreria wiringPi, ve el GPIO no esta listo, salimos de la aplicacion
if(wiringPiSetup()==-1){
    exit(1);
}

while(attempts){
    //Intentamos leer el valor del gpio, llamando a la funcion
    int success = dht11_read_val();

    //Si leemos con exito, salimos del while, y se acaba el programa
    if (success){
        break;
    }

    //Si no lee con exito, restamos 1, al numero de intentos
    attempts--;

    //Esperamos medio segundo antes del siguiente intento.
    delay(500);
}
return 0;
}
```

**Apêndice G – Script para testar comunicação serial com o Rpi**

```
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include <wiringPi.h>
#include <wiringSerial.h>

int main ()
{
    int fd ;
    int count ;
    unsigned int nextTime ;

    if ((fd = serialOpen ("/dev/ttyAMA0", 19200)) < 0)
    {
        fprintf (stderr, "Unable to open serial device: %s\n", strerror (errno)) ;
        return 1 ;
    }

    if (wiringPiSetup () == -1)
    {
        fprintf (stdout, "Unable to start wiringPi: %s\n", strerror (errno)) ;
        return 1 ;
    }

    nextTime = millis () + 300 ;

    for (count = 0 ; count < 256 ;)
    {
        if (millis () > nextTime)
        {
            printf ("\nOut: %3d: ", count) ;
            fflush (stdout) ;
            serialPuchar (fd, count) ;
            nextTime += 300 ;
            ++count ;
        }
    }
}
```

```
delay (3) ;

while (serialDataAvail (fd))
{
    printf (" -> %3d", serialGetchar (fd)) ;
    fflush (stdout) ;
}

printf ("\n") ;
return 0 ;
}
```

## Apêndice H – Script para coletar dados dos sensores analógicos

```
#include "SanUSB48.h"// https://www.youtube.com/watch?v=1B21b3zA4Ac
#pragma interrupt interrupção //Tem que estar aqui ou dentro do firmware.c
void interrupcao(){      }
float result0, result1, result2, result3, tensao, tensaoerro, corrente, hall, temp, tempsoma=0, tempfinal=0,
radiacao; //16 bits
unsigned char i=0;
int x;
const rom char str[10];

//-----FLOAT PARA SERIAL-----
void FloattoAscSerial(float value, char *string)
{
    if (value < 0) { // se valor for negativo
        *string++ = '-';
        value = -value;
    }
    sprintf(string, (const far rom char *) "%lu.%02u",
            (long) value, //int Decimal = diferenca do float e a parte inteira (long)
            (int) ((value - (long) value) * 100));
    sendsw((char *)str);
}

//-----LÊ-----
void main(){
clock_int_48MHz(); //Função necessaria para o dual clock
taxa_serial(19200);
habilita_canal_AD (AN0_a_AN3);

while(1){
    //-----LÊ-----
    result0 = le_AD10bits(0); tempo_ms(50); //tensão
    result1 = le_AD10bits(1); tempo_ms(50); //corrente
    result2 = le_AD10bits(2); tempo_ms(50); //temp LM35
    result3 = le_AD10bits(3); tempo_ms(50); //radiação
    //-----CONVERTE-----
    if (result0<=1024) { //só se for um valor aceitável
        tensao= (result0 * 4.28 * 4.55 )/1023;} //tensão
```

```

if (tensao>=10) { tensaoerro=tensao;} // valor estranho exclui
// if (tensao<=0) {tensao=0;}

if (result1<=1024){
corrente= (result1 * 4.28)/1023;} //corrente

if (corrente>=2.5){
hall=(corrente-2.5)/0.2; } //hall é a corrente com correção de offset de 2.5
//if (hall>=3.0){ hall=3;}
if (hall<=0.4) {hall=0.01;}

if (result2<=1024){
temp= (result2 * 428)/1023;} //graus LM35
for (x=0;x<10;x++){
    tempsoma=tempsoma+temp;
    tempo_ms(20);
}
tempfinal=tempsoma/10;
tempsoma=0;
// if (tempfinal<=0) {tempfinal=0;}

radiacao= (result3 * 250.4 * 5 )/1023;
if (radiacao<=70) {radiacao=0.01;}

//-----ENVIA-----
swputc('a');
FloattoAscSerial(tensao,str);
swputc('b');
FloattoAscSerial(hall,str);
swputc('e');
FloattoAscSerial(tempfinal,str);
swputc('f');
FloattoAscSerial(radiacao,str);
swputc('!'); //Indica fim do valor
    inverta_saida(pin_b7);
    tempo_ms(2000);

}
}

```

## Apêndice I – *Script* para receber os dados da serial e enviar a Internet

```

#include <stdio.h> // http://sanusb.org/monitor/
#include <string.h>
#include <errno.h>
#include <wiringPi.h>
#include <wiringSerial.h>
#include <stdlib.h>
#include <string.h>
#include <stdint.h>
#include <unistd.h>
#include <termios.h>
#include <fcntl.h>
#include <time.h>

char charRecebido;
int fd;
int flagt=0, flagb=0, flagc=0, flagd=0, flage=0, flagf=0, flagg=0, flagh=0, flagfim=0;
char x;
char str[10];
int i=0; // a: tension b: corriente c: temperatura2 d: humedad e: temperatura1 f: radiacion
const char host[] = "";
char *str1="", stra[10]="", strb[10]="", strc[10]="", strd[10]="", stre[10]="", strf[10]="", strg[10]="",
strv[10]="";

int getch(void)
{
    int ch;
    struct termios oldt, newt;
    long oldf, newf;

    tcgetattr(STDIN_FILENO, &oldt);
    newt = oldt;
    newt.c_lflag &= ~(ICANON | ECHO);
    tcsetattr(STDIN_FILENO, TCSANOW, &newt);
    oldf = fcntl(STDIN_FILENO, F_GETFL, 0);
    newf = oldf | O_NONBLOCK;
    fcntl(STDIN_FILENO, F_SETFL, newf);
    ch = getchar();
    fcntl(STDIN_FILENO, F_SETFL, oldf);

```

```

    tcsetattr(STDIN_FILENO, TCSANOW, &oldt);
    return ch;
}

PI_THREAD (recebe_serial) {
    (void)piHiPri (10) ;
    int i = 0, z=0;
    char temp;
    char str1[10];

    /*******
    // Leer fichero DHT11 *
    /*******

        FILE *fr;
        char strhumedade[10];
        char strhumedade1[10];

        char strtemp_ambiente[10];
        char strtemp_ambiente1[10];
        int humedad, humedad1;
        int temp_ambiente, temp_ambiente1;
        int fechaHora;
        fr = fopen ("/home/share/dht11.log", "r");
        while (feof(fr) == 0)
            {
                fseek( fr, -15L, SEEK_END );
                fscanf  (fr,  "%d,%d,%d,%d,%d\n",  &fechaHora,  &humedad,  &humedad1,
&temp_ambiente, &temp_ambiente1);
                sprintf(strhumedade, "%d", humedad);
                sprintf(strhumedade1, "%d", humedad1);
                strcat(strhumedade, ".");
                strcat(strhumedade, strhumedade1);
                printf("humedad : %s\n", strhumedade);
                sprintf(strtemp_ambiente, "%d", temp_ambiente);
                sprintf(strtemp_ambiente1, "%d", temp_ambiente1);
                strcat(strtemp_ambiente, ".");
                strcat(strtemp_ambiente, strtemp_ambiente1);
                printf("temperatura_ambiente : %s\n", strtemp_ambiente);
            }
}

```

```

        fclose(fr);
//*****
// Recibir datos de la serial*
//*****
while(1) {
    charRecebido = serialGetchar(fd);

    // a: tension b: corriente c: temperatura2 d: humedad e: temperatura1 f: radiacion g: presion h: vazao
    if (flagt==1 && flagb==0 && charRecebido != 'b'){
        sprintf(str1, "%c", charRecebido);
        strcat(stra, str1);
        printf("tensao = %s \r\n", stra); //debug
    }

    if (flagb==1 && flagf==0 && charRecebido != 'f'){
        sprintf(str1, "%c", charRecebido);
        strcat(strb, str1);

printf("corrente = %s \r\n", strb); //debug
    }

    if (flagf==1 && flagim==0 && charRecebido != '!'){
        sprintf(str1, "%c", charRecebido);
        strcat(strf, str1);
        printf("radiacao = %s \r\n", strf); //debug
    }

switch(charRecebido)
{
    case 't': //printf("passou em a \r\n");
        flagt=1;flagf=0;flagim=0;
        break;

    case 'b': //printf("passou em b\r\n");
        flagb=1;flagt=0;
        break;

    case 'f': //printf("passou em f\r\n");
        flagf=1;flagb=0;
        break;
}
}

```

```

case '!': //printf("passou em !\r\n");
        flagfim=1; flagf=0;
        break;
    }

    fflush(stdout);
//*****
// Enviar para Internet    *
//*****

    if (flagfim==1){
        time_t tiempo = time(0);
        struct tm *tlocal = localtime(&tiempo);
        struct tm result;
        char date[100], time[100];
        char fechaOK[128]="";
        strftime(date,128,"%Y-%m-%d",tlocal);
        strftime(time,128,"%H:%M:%S",tlocal);
        strcat(fechaOK, date);
        strcat(fechaOK, "%20");
        strcat(fechaOK, time);
        printf ("Hoy es: %s\n", fechaOK);

        char strPost[200] = "curl \"http://";
        strcat(strPost, host);
        strcat(strPost, "/ftpmonitor/getParameterFromRPi.php?action=send1");
        strcat(strPost, "&voltage=");
        strcat(strPost, stra);
        strcat(strPost, "&current=");
        strcat(strPost, strb);
        strcat(strPost, "&temperature2=");    // C
        strcat(strPost, strtemp_ambiente);
        strcat(strPost, "&humidity=");    // D
        strcat(strPost, strhumedade);
        strcat(strPost, "&radiation=");
        strcat(strPost, strf);
        strcat(strPost, strg);
        strcat(strPost, "&flowrate=");
        char fechaWeb[128]="";
        strcpy(fechaWeb, fechaOK);
    }

```

```

strcat(strPost, "&date=");
strcat(strPost, fechaOK);
strcat(strPost, "\\");
printf (strPost);
system(strPost);

        strcpy(stra,"");strcpy(strb,"");strcpy(strc,""); //limpa as strings
        strcpy(strd,"");
        strcpy(strf,"");
        strcpy(strg,"");
        strcpy(strv,"");
        break;
    }
}
}
int main(void){
// Visualizados em: //
https://docs.google.com/spreadsheets/d/1tDX1NaMiKmKVawsHfS7hpAf7lB5ZeHpYu0nLmau-TE/edit#gid=1687639411
char str1[150] = "curl
'https://docs.google.com/forms/d/1PZOqjnitER0m03Ix4r9gDBqhp7Xs1YrPmjLymE2VWAU/formResponse?ifq&entry.962023089=SerialConfigTest&submit=Submit';
if ((fd = serialOpen ("/dev/ttyAMA0", 19200)) < 0){
    fprintf (stderr, "Unable to open serial device: %s\n", strerror (errno)) ;
    return 1 ;
}
wiringPiSetup() ;
pinMode(1, OUTPUT) ;
piThreadCreate (recebe_serial) ;
digitalWrite (1, HIGH);
strcat(str1,">/dev/null 2>&1"); //quiet the response
system(str1);
fflush(stdout);
fflush(stdout);
printf("\n\nEstablished serial connection!!!\n");
fflush(stdout);
    delay (2000);
    delay(5000);
    x = getch();
return 0;}

```

## Apêndice J – Script PHP para inserir os dados no banco de dados MySQL na nuvem

```

<?php

ini_set('default_charset','UTF-8');
header('Content-Type: text/html; charset=utf-8');
header("Refresh:5"); // Refrsh after 5 seconds
require_once ('path.php');
include(INCLUDES_PATH.'connection.php');

if (isset($_GET['action']))
    {
        $action = $_GET['action'];
        // Get parameters
        /*******
        $voltage = $_GET['voltage']; // Sensor 1
        $current = $_GET['current']; // Sensor 2
        $temperature2 = $_GET['temperature2']; // Sensor 3
        $humidity = $_GET['humidity']; // Sensor 4
        $radiation = $_GET['radiation']; // Sensor 6
        $pressure = $_GET['pressure']; // Sensor 7
        $flowrate = $_GET['flowrate']; // Sensor 8
        $temperature_other = $_GET['temperature_other']; // Sensor 8
        $temperature1 = $_GET['mytemperature']; // Sensor 5
        $date = $_GET['date'];
        $pt100 = $_GET['pt100'];

if ($action == "send1")
    {
        if(is_numeric(floatval($voltage)))
            {
                $sql_voltage = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
                    VALUES ('$date', '$date', '$voltage', 1)";
            }
        else
            {
                $sql_voltage = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
                    VALUES ('$date', '$date', 0, 1)";
            }
        }

        if(is_numeric(floatval($current)))
            {
                $sql_current = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
                    VALUES ('$date', '$date', '$current', 2)";
            }
        else
            {
                $sql_current = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
                    VALUES ('$date', '$date', 0, 2)";
            }
        }

        if(is_numeric(floatval($temperature2)))
            {
                $sql_temperature2 = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
                    `sensor`) VALUES ('$date', '$date', '$temperature2', 3)";
            }
        else
            {

```

```

        $sql_temperature2 = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
`sensor`) VALUES ('$date', '$date', 0, 3);
    }
    if(is_numeric(floatval($humidity)))
    {
        $sql_humidity = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
`sensor`) VALUES ('$date', '$date', '$humidity', 4);
    }
    else
    {
        $sql_humidity = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
`sensor`) VALUES ('$date', '$date', 0, 4);
    }
    if(is_numeric(floatval($radiation)))
    {
        $sql_radiation = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
`sensor`) VALUES ('$date', '$date', '$radiation', 6);
    }
    else
    {
        $sql_radiation = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`,
`sensor`) VALUES ('$date', '$date', 0, 6);
    }

$query_current = mysql_query($sql_current);
$query_voltage = mysql_query($sql_voltage);
$query_radiation = mysql_query($sql_radiation);
$query_temperature2 = mysql_query($sql_temperature2);
$query_humidity = mysql_query($sql_humidity);

if($query_current && $query_voltage && $query_radiation && $query_temperature2 && $query_humidity)
{
    echo "Tension, corriente, temperatura ambiente, humedad y radiacion";

    echo "voltage ".$query_voltage;
    echo $query_current;
    echo $query_temperature2;
    echo $query_humidity;
    echo "radiation ". $query_radiation;

    echo $date;
}
else
{
    die (mysql_error());
}

if(is_numeric(floatval($pt100)))
{
    $sql_pt100 = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
VALUES ('$date', '$date', '$pt100', 5);
}
else
{
    $sql_pt100 = "INSERT INTO monitoramento2 (`dataMedida`, `dataInput`, `valor`, `sensor`)
VALUES ('$date', '$date', 0, 5);
}

```

```

echo "pt100 ".$pt100;

$query_pt100 = mysql_query($sql_pt100);

if ($query_pt100)
    {
        echo "OK";
        echo "pt100 ".$pt100;
        echo $date;
    }
else
    {
        die (mysql_error());
    }
}

else if ($action == "send2")
    {
        if(is_numeric(floatval($pressure)))
            {
                $sql_pressure = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', '$pressure', 7)";
            }
            else
            {
                $sql_pressure = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', 0, 7)";
            }

            if(is_numeric(floatval($flowrate)))
            {
                $sql_flowrate = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', '$flowrate', 8)";
            }
            else
            {
                $sql_flowrate = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', 0, 8)";
            }

            if(is_numeric(floatval($temperature_other)))
            {
                $sql_temperature_other = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', '$temperature_other', 9)";
            }
            else
            {
                $sql_temperature_other = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', 0, 9)";
            }
            if(is_numeric(floatval($temperature1)))
            {
                $sql_temperature1 = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', '$temperature1', 5)";
            }
            else
    }

```

```

        {
            $sql_temperature1 = "INSERT INTO monitoramento2
(`dataMedida`, `dataInput`, `valor`, `sensor`) VALUES ('$date', '$date', 0, 5)";
        }

$query_pressure = mysql_query($sql_pressure);
$query_flowrate = mysql_query($sql_flowrate);
$query_temperature_other = mysql_query($sql_temperature_other);
$query_temperature1 = mysql_query($sql_temperature1);

if($query_pressure && $query_flowrate && $query_temperature_other &&
$query_temperature1)
    {
        echo "OK";

        echo "pressure ".$pressure;
        echo "<br>";
        echo "flowrate ".$flowrate;
        echo "<br>";
        echo "temperature_other ".$temperature_other;
        echo "<br>";
        echo "temperature1 ".$temperature1;
        echo $date;
    }
else
    {
        die (mysql_error());
    }
}

}

// close mysql
mysql_close();

?>

```