

TRABALHO DE CONCLUSÃO DE CURSO

**SISTEMAS DE MEDIÇÃO DE ENERGIA ELÉTRICA VISANDO
APLICAÇÃO EM IDENTIFICAÇÃO E PREVENÇÃO DE FALHAS**

Caroline Cunha Barbosa

Brasília, Julho de 2018

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

TRABALHO DE CONCLUSÃO DE CURSO

**SISTEMAS DE MEDIÇÃO DE ENERGIA ELÉTRICA VISANDO
APLICAÇÃO EM IDENTIFICAÇÃO E PREVENÇÃO DE FALHAS**

Caroline Cunha Barbosa

*Relatório submetido ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Bacharel em Engenharia Elétrica*

Banca Examinadora

Prof. Adolfo Bauchspiess, ENE/UnB
Orientador

Prof. Lélío Ribeiro Soares Júnior, ENE/UnB
Examinador Interno

Eng. André Luiz Siega Nepomuceno
Examinador Externo

Dedicatória

À minha vó Shirley (In Memoriam) por acreditar no meu potencial mesmo quando eu duvidei.

Caroline Cunha Barbosa

Agradecimentos

Não há atalhos na vida. Para alcançar algo que se imaginava impossível é necessário crescer para se tornar a pessoa que realize o impossível. Algumas pessoas contribuíram nessa longa trajetória.

Primeiramente, gostaria de agradecer à mamãe Sheila e ao papai Wagner por apresentarem todos os caminhos possíveis de seguir mas permitirem que eu tomasse a decisão final e me apoiarem não importasse qual fosse a escolha.

Agradeço à minha família por sempre comemorarem comigo as minhas conquistas e me levantarem nas derrotas. Obrigada pela compreensão pelas vezes em que fiquei indisponível para correr atrás dessa realização.

Agradeço a UnB por me proporcionar essa incrível etapa da minha vida. Especialmente, meu companheiro desde o primeiro dia do curso, Gustavo Oliveira, que sempre acreditou no sonho. Agradeço a equipe da EREKO. A professora Dianne que mesmo sem saber me motivou a seguir na engenharia. Ao Matheus por buscar uma vida épica e me levar junto nesta jornada.

Agradeço a meu orientador, professor Adolfo, pela paciência e por toda ajuda ao longo deste trabalho.

Agradeço aos meus amigos de curso e de UnB por todas as experiências compartilhadas, pelas risadas e pela ajuda nos momentos mais desesperadores. Obrigada Antônio, Débora, Kássia, Marina, Sidney e Túlio por me ajudar a segurar essa barra.

Por fim mas não menos importante, gostaria de agradecer aos amigos de intercâmbio que mesmo distantes estão sempre presentes e tentam inspirar uns aos outros na busca do sucesso e de cookies. Obrigada Aline, Anna, Bruma, César, Dr. Cookies, Luis, Matheus, Paulo Mallmann, Pedro, Rafaela e Túlio José.

Caroline Cunha Barbosa

RESUMO

O presente texto apresenta um sistema de monitoramento de consumo de energia elétrica, em tempo real, em que foram feitos estudos para a identificações de falhas em sistemas térmicos. A realização de medições procura colaborar para a eficiência energética em aplicações de ambientes prediais. O trabalho segue uma linha de pesquisa do Laboratório de Robótica e Automação da Universidade de Brasília em que um módulo medidor de energia foi construído anteriormente. A partir deste foi criada uma rede demonstrativa abrangendo três módulos integrados a um sistema de aquisição de dados aberto, ScadaBR. Visando a mobilidade deste sistema para diagnósticos de eficiência energética em prédios e residências foi adotada uma abordagem sem fio dos componentes de modo a oferecer mobilidade aos componentes e não restringir a um local de análise. Os módulos são responsáveis por monitorar os parâmetros relacionados a energia elétrica, um módulo central para recepção dos dados transmitidos dos outros módulos e um módulo de determinação da temperatura ambiente. Por fim, foram realizados ensaios para validação de identificação de falhas do sistema.

Palavras chave: Medição, energia elétrica, detecção de falha, aquisição de dados, ScadaBR.

ABSTRACT

The present text presents a system that monitor the consumption of electric energy, in real time, in which studies were made for the identification of failures in thermal systems. The measurements seek to collaborate for energy efficiency in residential applications. The work follows a line of research of the Laboratory of Automation and Robotics of the University of Brasília in which an energy meter module was constructed previously. Based on this, a demonstrative network was created containing three modules integrated to an open data acquisition system, ScadaBR. Aiming the mobility of this system for diagnostics of energy efficiency in buildings and residences, a wireless approach was adopted in order to offer mobility to components and not restrict to a place of analysis. The modules are responsible for monitoring the electrical parameters, a central module for receiving the data from other module and a module for determining the ambient temperature. Finally, tests were performed to validate the identification of failures.

Keywords: Measurement, electrical energy, fault detection, data acquisition, ScadaBR.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONTEXTUALIZAÇÃO	1
1.1.1	A ESCASSEZ DE ENERGIA	1
1.1.2	EFICIÊNCIA ENERGÉTICA E ECONOMIA DE ENERGIA	2
1.2	MOTIVAÇÃO	2
1.3	DESCRIÇÃO DO PROBLEMA	3
1.4	OBJETIVO DO PROJETO	3
1.4.1	OBJETIVOS ESPECÍFICOS	4
1.5	APRESENTAÇÃO DO MANUSCRITO	4
2	FUNDAMENTAÇÃO TEÓRICA	5
2.1	AUTOMAÇÃO PREDIAL	5
2.2	PROTOCOLO MODBUS	6
2.3	FUNDAMENTOS DE MEDIÇÃO DE ENERGIA ELÉTRICA	6
2.3.1	POTÊNCIA INSTANTÂNEA	6
2.3.2	VALOR EFICAZ	7
2.3.3	POTÊNCIA MÉDIA	7
2.3.4	FATOR DE POTÊNCIA	8
2.3.5	POTÊNCIA COMPLEXA	8
2.3.6	ENERGIA ELÉTRICA	9
2.4	APARELHOS DE MEDIÇÃO	9
2.4.1	VOLTÍMETRO	9
2.4.2	AMPERÍMETRO	9
2.4.3	WATTÍMETRO	9
2.4.4	MEDIDOR DE ENERGIA ELÉTRICA	10
2.5	TEMPO REAL	10
2.6	PRECISÃO E CALIBRAÇÃO	11
3	METODOLOGIA	12
3.1	CALIBRAÇÃO	12
3.2	AMBIENTE DE ESTUDO	13
3.3	EQUIPAMENTOS UTILIZADOS	13
3.4	PLANO DE ENSAIOS ADOTADO	15

3.4.1	ENSAIO DE LONGA DURAÇÃO	15
3.4.2	ENSAIO COM MÚLTIPLOS ACIONAMENTOS DA CARGA.....	15
3.4.3	CARGA COM OPERAÇÃO PADRÃO	15
3.4.4	ENSAIO COM SAÍDA DE AR PARCIALMENTE BLOQUEADA	16
3.4.5	ENSAIO SAÍDA DE AR TOTALMENTE BLOQUEADA	16
3.5	ANALISE DO ERRO EXPERIMENTAL	16
4	DESENVOLVIMENTO	17
4.1	VISÃO GERAL DO PROJETO	17
4.1.1	SISTEMA PROPOSTO	17
4.2	MÓDULO MEDIDOR DE PARÂMETROS ELÉTRICOS	18
4.2.1	SUBMÓDULO DE MEDIÇÃO	18
4.2.2	ARDUINO	21
4.2.3	CIRCUITO COMUNICAÇÃO	22
4.2.4	LOGICA DE PROGRAMAÇÃO	22
4.3	MÓDULO MEDIDOR DE TEMPERATURA	23
4.3.1	SENSOR DE TEMPERATURA	23
4.3.2	PROCESSAMENTO DO MÓDULO	24
4.4	SCADA	25
5	RESULTADOS EXPERIMENTAIS	29
5.1	CALIBRAÇÃO E AJUSTES	29
5.2	ENSAIO DE LONGA DURAÇÃO COM MÓDULO MEDIDOR DE ENERGIA.....	29
5.3	ENSAIO COM MÚLTIPLOS ACIONAMENTOS DA CARGA EM DIFERENTES MODOS DE OPERAÇÃO	31
5.4	TESTES PARA DETECÇÃO DE FALHAS	32
5.4.1	CARGA COM OPERAÇÃO PADRÃO	32
5.4.2	ENSAIO COM SAÍDA DE AR PARCIALMENTE BLOQUEADA	33
5.4.3	ENSAIO SECADOR DE CABELO COM SAÍDA DE AR TOTALMENTE FECHADA DU- RANTE FUNCIONAMENTO	34
5.4.4	ENSAIO SECADOR DE CABELO EM TENSÃO 110V	35
6	CONCLUSÕES E PERSPECTIVAS	37
6.1	TRABALHOS FUTUROS	37
	REFERÊNCIAS BIBLIOGRÁFICAS.....	39
	ANEXOS.....	41
I	DIAGRAMAS ESQUEMÁTICOS	42
II	TABELAS DOS COMPONENTES.....	44
III	CÓDIGOS	45
III.1	CÓDIGO UTILIZADO PARA MEDIDOR DE TEMPERATURA	45

LISTA DE FIGURAS

1.1	Selo Procel de Economia de Energia(PROCELINFO, 2015).....	2
1.2	Objetivos de Desenvolvimento Sustentável da ONU(NAM, 2015)	3
2.1	Sistema de automação predial(EUBAC, 2017).....	5
2.2	Arquitetura de MODBUS RTU(MODBUS, 2014).....	6
2.3	Modelo de Wattímetro analógico(POLITERM, 2016)	10
2.4	Modelo de medidor de energia elétrica residencial(SABERELETRICA, 2016).....	10
3.1	Laboratório de Automação e Robótica (LARA/UnB).....	13
3.2	Sala do LARA/UnB.....	13
3.3	Secador de cabelo convencional	14
3.4	modelo de instalação do módulo medidor.....	14
4.1	Modelo simplificado do sistema proposto	18
4.2	Estrutura do módulo	18
4.3	Módulo medidor de energia utilizado no trabalho.....	19
4.4	Circuito de alimentação(NEPOMUCENO; COZENDEY, 2013)	19
4.5	Circuito de medição de tensão, corrente e potência, utilizando um SA9903B(NEPOMUCENO; COZENDEY, 2013)	20
4.6	Ligação entre a saída de dados medidos com a entrada do processamento(NEPOMUCENO; COZENDEY, 2013)	21
4.7	Modelo simplificado do sistema proposto.....	22
4.8	Módulo XBee	22
4.9	Sensor de Temperatura	23
4.10	Módulo de medição da temperatura.....	24
4.11	Estrutura do programa do medidor da temperatura	25
4.12	Interface do sistema ScadaBR.....	26
4.13	<i>data sources</i> criados para o supervisor desenvolvido	26
4.14	<i>data points</i>	26
4.15	Detalhes de um <i>data point</i>	27
4.16	Estrutura do medidor da temperatura	28
5.1	Gráficos dos parâmetros elétricos ao longo do período de ensaio de longa duração	30
5.2	Gráficos dos parâmetros elétricos ao longo do ensaio de acionamentos alternados	31
5.3	Gráficos dos parâmetros elétricos ao longo do período do ensaio padrão	32

5.4	Gráficos comparativo dos parâmetros elétricos no ensaio com saída de ar parcialmente bloqueada.....	33
5.5	Gráficos dos parâmetros elétricos do ensaio com saída de ar obstruída.....	34
5.6	Gráficos comparativo dos parâmetros elétricos do ensaio com obstrução total da saída de ar.	35
I.1	Esquemático Módulo de temperatura.....	42
I.2	Esquemático de aplicação sugerida do SA9903B da SAMES.....	43

LISTA DE TABELAS

3.1	Especificações técnicas da carga utilizada.....	14
II.1	Tabela para o circuito de aplicação sugerida do SA9903B da SAMES	44

LISTA DE SÍMBOLOS

Símbolos Latinos

$v(t)$	Tensão instantânea	[V]
$i(t)$	Corrente instantânea	[A]
$p(t)$	Potência instantânea	[VA]
T	Período	[s]
X	Variável	
t	Tempo	[s]
t_o	Tempo inicial	[s]
P	Potência ativa	[W]
fp	Fator de potência	
V	Tensão	[V]
I	Corrente	[A]
S	Potência Complexa	[VA]
Q	Potência reativa	[var]
E	Energia	[Wh]

Símbolos Gregos

ω	Frequência angular	[rad/s]
Δ	Varição entre duas grandezas similares	
θ	Ângulo de fase	[rad]
α	Ângulo de fase da tensão	[rad]
β	Ângulo de fase da corrente	[rad]
τ	Temperatura	[°C]

Subscritos

<i>amb</i>	Ambiente
<i>ext</i>	Externo
<i>in</i>	Entrada
<i>ex</i>	Saída
<i>m</i>	Máximo
<i>RMS</i>	<i>Root Mean Square</i>

Sobrescritos

.	Variação temporal
—	Valor médio

Siglas

ABNT	Associação Brasileira de Normas Técnicas
BAS	<i>Building Automation System</i>
CA	Corrente Alternada
CC	Corrente Contínua
CI	Circuito Integrado
CS	<i>Chip Select</i>
EPE	Empresa de Pesquisa Energética
FMO	<i>Mains Voltage Sense Zero</i>
ISO	<i>International Organization for Standardization</i>
LARA	Laboratório de Robótica e Automação
MISO	<i>Master Input Slave Output</i>
MME	Ministério de Minas e Energia
MOSI	Master Output Slave Input
PROCEL	Programa Nacional de Conservação de Energia Elétrica
RTU	<i>Remote Terminal Unit</i>
SCK	<i>Serial Clock</i>
SDO	<i>Serial Data Out</i>
SPI	<i>Serial Peripheral Interface</i>
TCP	<i>Transmission Control Protocol</i>

Capítulo 1

Introdução

1.1 Contextualização

Com o crescente desenvolvimento econômico e populacional, ocorre um aumento na demanda por energia elétrica. Além disso, o setor elétrico tem que considerar a disponibilidade de recursos, fatores ambientais e econômicos para a geração da energia necessária para abastecer todos os segmentos da indústria, comércio e a população. Tendo isso em mente, alguns programas governamentais têm o papel de incentivar o uso eficiente da energia elétrica.

1.1.1 A escassez de energia

A energia elétrica está presente no cotidiano das pessoas em quase todas as atividades, seja no trabalho ou nas residências. Devido ao consumo energético contínuo e crescente, há a necessidade de ampliação da oferta de energia elétrica.

A matriz energética mundial tem como principal fonte de energia elétrica os combustíveis fósseis (BRONZATTI; IAROSINSKI, 2008). O uso de combustíveis fósseis para geração de energia elétrica é classificado como uma fonte não renovável, ou seja, é uma fonte energética que exige recursos que levam milhões de anos para serem formados. Visto que o processo de formação dos recursos demora tempo demasiadamente longo, estes são limitados. Além disso, a queima de combustíveis fósseis emite gases que alteram as condições climáticas.

No Brasil, a matriz energética tem como principal fonte, a energia hidrelétrica, sendo esta uma fonte renovável. As fontes renováveis têm como principais recursos elementos que se regeneram em um período curto e, portanto, tem uma disponibilidade por um longo período, tais como as águas de um rio ou o sol. Como as usinas hidrelétricas não são construídas rapidamente, a ampliação de geração de energia elétrica limpa é lenta e dispendiosa.

Em 2015 os níveis dos reservatórios das hidrelétricas estavam baixos, devido à falta de chuvas. Neste período, a energia de hidrelétricas não estava em abundância e, portanto, a energia de termelétricas foi requisitada para suprir a demanda. Por utilizar combustíveis fósseis, cujo preço é mais caro, o valor nas

contas de energia elétrica subiu.

Tendo conhecimento de que com o avanço tecnológico e o desenvolvimento dos países emergentes, há a necessidade de aumentar a disponibilidade de energia elétrica e de utilizar da maneira mais eficiente possível. Ademais, deve-se considerar os aspectos financeiros e ambientais na questão do setor energético.

1.1.2 Eficiência energética e economia de energia

O uso consciente dos recursos mostra-se uma forma eficiente de gerenciamento econômico e demonstra como o desenvolvimento tecnológico está aliado com a preservação destes. Analisando o setor elétrico é possível notar a presença do Programa Nacional de Conservação de Energia Elétrica (PROCEL), criado pelo Ministério de Minas e Energia (MME) para definir estratégias de incentivo quanto ao uso responsável e eficiente de energia elétrica.

O PROCEL é um indicativo de que é necessário implementar medidas de eficiência energética com o objetivo de economizar recursos e melhorar o emprego destes. Dentre as medidas implementadas, encontram-se os selos Procel de Economia de Energia, figura 1.1, os quais visam informar os equipamentos mais eficientes e que consomem menos energia (PROCELINFO, 2015).



Figura 1.1: Selo Procel de Economia de Energia(PROCELINFO, 2015)

Elevados índices de desperdícios aumentam a necessidade de energia no mercado e, portanto, aumentam os preços nas tarifas. Com medidas de fácil implementação é possível diminuir os desperdícios e, deste modo, economizar (KOPKE; ZLOCCOWICK; SCHREIER, 2015).

De acordo com (MARQUES; HADDAD; GUARDIA, 2007), um sistema de refrigeração pode chegar a consumir 60% do total de energia elétrica de um edifício comercial. Com a utilização de equipamentos mais eficientes ou o controle do sistema pode-se melhorar as condições de conforto do edifício e economizar energia.

Com a economia de energia, busca-se o adiamento da necessidade de ampliação de unidades geradoras para o sistema elétrico e se contribui para a preservação dos recursos naturais. Além de que a energia mais barata é aquela que não precisou ser consumida.

1.2 Motivação

De acordo com a plano de ação da Organização das Nações Unidas (ONU) sobre o desenvolvimento sustentável (NAM, 2015) , existem 17 objetivos a serem cumpridos até 2030, os quais visam integrar o de-

envolvimento econômico, social e ambiental de forma sustentável e balanceada. Dentre estes, os objetivos energia acessível e limpa, cidades e comunidades sustentáveis e consumo e produção responsáveis foram as motivações deste trabalho no que tange ao incentivo da melhora na eficiência energética e no melhor gerenciamento dos recursos naturais.



Figura 1.2: Objetivos de Desenvolvimento Sustentável da ONU(NAM, 2015)

1.3 Descrição do problema

Na busca pelo aumento na produtividade no ambiente de trabalho, certas condições de conforto devem ser praticadas. Até mesmo em residências, dependendo da localidade, um certo nível de conforto é necessário para a realização das atividades domésticas e de lazer. As condições climáticas fazem com que seja necessário a presença de aparelhos de refrigeração em praticamente todos os ambientes de trabalho e em muitas residências. Com isto, deduz-se que há uma grande demanda por energia elétrica, visando a manutenção da temperatura para o conforto térmico em ambientes fechados.

A existência de apenas um medidor de energia elétrica na entrada do fornecimento do edifício ou casa, impossibilita determinar o real consumo de cada equipamento, individualmente, do edifício ou da residência. Uma forma de individualizar o monitoramento dos equipamentos é a instalação de medidores exclusivos para cada equipamento, o qual será analisado para a melhora do consumo.

O sistema proposto para determinar o real consumo de cada equipamento consiste em instalar medidores de consumo elétrico em cada ambiente em que se deseja realizar o estudo para o diagnóstico de consumo.

1.4 Objetivo do projeto

Aplicar com o auxílio de um sistema supervisor e de aquisição de dados um módulo de medição de energia elétrica que propicie a identificação e prevenção de falhas em tempo real com mobilidade.

A realização de medições visa utilizar os recursos disponíveis da forma eficiente procurando detectar e prever falhas no sistema. Para o sistema supervisor aberto escolhido, ScadaBr(SCADABR, 2014), deverá ser criada uma rede demonstrativa abrangendo três módulos integrados ao supervisor. Dois módulos são responsáveis por monitorar os parâmetros relacionados a energia elétrica e um módulo de determinação da temperatura ambiente.

1.4.1 Objetivos Específicos

O trabalho utilizará um módulo medidor de energia elétrica desenvolvido pelo trabalho de (??) e a programação desenvolvida no mesmo.

- Fazer análise dos módulos existentes com relação ao funcionamento e aquisição de dados;
- Implementar comunicação sem fio entre os módulos e uma central de monitoramento, utilizando XBEE;
- Adaptar os códigos utilizados para a implementação dos módulos;
- Elaborar um sistema supervisor, utilizando o ScadaBR, para aquisição e monitoramento dos dados por meio do desenvolvimento de um banco de dados para este fim;
- Realizar calibração dos módulos responsáveis pelos parâmetros de energia elétrica;
- Definir um local para realização dos ensaios experimentais;
- Definir os equipamentos que serão utilizados para monitoramento;
- Realizar ensaio utilizando um secador de cabelo para a verificação do funcionamento do sistema desenvolvido;

1.5 Apresentação do manuscrito

No capítulo 2 é feita uma revisão bibliográfica sobre o tema de estudo. Em seguida, o capítulo 4 descreve os recursos e a metodologia empregada no desenvolvimento do projeto. Na continuidade, o capítulo 3 apresenta as estratégias utilizadas para a realização dos experimentos de análise. Os resultados experimentais são discutidos no capítulo 5, seguido das conclusões e discussões no capítulo 6. Os anexos contém material complementar, tais como os esquemáticos e códigos utilizados no desenvolvimento deste trabalho.

Capítulo 2

Fundamentação teórica

2.1 Automação predial

Automação é um sistema que visa operar processos por meio do controle de dispositivos de modo a otimizar o trabalho. Ao receber informações de um equipamento, o sistema de automação toma decisões a serem executadas pelo equipamento. A automação aplicada aos prédios surgiu da busca pelo gerenciamento dos sistemas prediais de forma inteligente. A automação predial, ou *Building Automation System*(BAS) tem o objetivo de facilitar o dia-a-dia das pessoas fornecendo conforto, segurança e economia.

Os equipamentos de um edifício são conectados a uma rede que faz o gerenciamento dos recursos do prédio, tais como o sistema de segurança, sistema de refrigeração, iluminação, entre outros. A figura 2.1 apresenta alguns sistemas que são monitorados por BAS.



Figura 2.1: Sistema de automação predial(EUBAC, 2017)

As funções do BAS são controlar o ambiente do edifício, operar o sistema de acordo com a demanda e ocupação, monitorar e realizar correções no sistema quando necessário. Para realizar todas estas funções é necessário a instalação de sensores, controladores e um protocolo de comunicação. O investimento nos equipamentos necessários para formar os sistema de automação predial é pequeno comparado com a economia proporcionada a longo prazo.

De acordo com (WANG, 2009), um prédio com um sistema de automação consegue ter uma redução

de consumo energético de até 30%.

Por meio da melhor aplicabilidade dos recursos utilizados em um prédio, é possível economizar energia, reduzir de custos de operação e reduzir a emissão de CO₂.

2.2 Protocolo MODBUS

O protocolo MODBUS visa assegurar uma comunicação mestre-escravo entre dispositivos. A comunicação pode ser entre sensores, dispositivos, instrumentos, computadores entre outros de modo a criar um ambiente onde seja possível monitorar e controlar os aparelhos (MODBUS, 2014). A figura 2.2 apresenta um exemplo de arquitetura de um protocolo MODBUS.

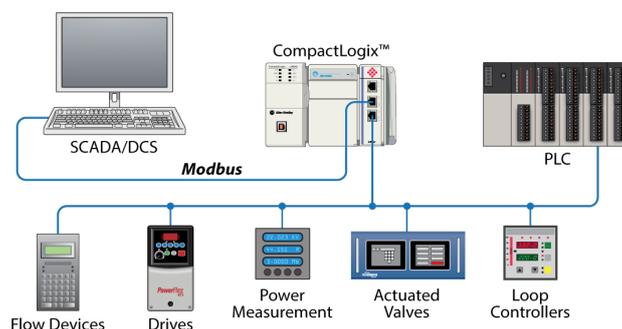


Figura 2.2: Arquitetura de MODBUS RTU(MODBUS, 2014)

Aplicações com *Remote Terminal Unit* (RTU) são possíveis com o uso do protocolo, visto que nestas a comunicação *wireless* é necessária ao supervisor do sistema.

2.3 Fundamentos de medição de energia elétrica

Por que medir o consumo de energia elétrica? A medição de sistemas elétricos tem muitas utilidades. A aplicação mais comum é para a cobrança da energia consumida da rede elétrica. Saber quanta potência está sendo gerada por uma instalação de painéis solares para consumo próprio ou quanto um motor em funcionamento está consumindo também são exemplos de aplicações de medidores. A medição dos parâmetros elétricos possibilita inspecionar com detalhes o consumo de uma instalação e identificar a ocorrência de desvios. Portanto, a medição fornece detalhes do consumo e economia de energia. A seguinte explicação dos parâmetros elétricos foi baseado em (IRWIN; NELMS, 2000)

2.3.1 Potência instantânea

A rede de transmissão elétrica fornece aos consumidores tensões CA. Deste modo, a representação de tensões e correntes é feita por meio de ondas senoidais. As equações 2.1 e 2.2, indicam as funções harmônicas da tensão instantânea e corrente instantânea, respectivamente.

$$v(t) = V_m \cos(\omega t) \quad (2.1)$$

$$i(t) = I_m \cos(\omega t - \theta) \quad (2.2)$$

Na expressão, V_m e I_m são as amplitudes, ou valor máximo. ω é a frequência angular dada em radianos por segundo e θ é o ângulo de fase da corrente dado comumente em radianos.

A potência instantânea absorvida por um dispositivo é definida como o produto da tensão instantânea pela corrente instantânea. Fazendo o produto das equações 2.1 e 2.2 tem-se

$$p(t) = v(t)i(t) \quad (2.3)$$

$$p(t) = V_m I_m \cos(\omega t) \cos(\omega t - \theta) \quad (2.4)$$

Utilizando identidades trigonométricas é possível simplificar a equação 2.4 para a seguinte forma:

$$p(t) = \frac{V_m I_m}{2} [\cos(\theta) + \cos(2\omega t - \theta)] \quad (2.5)$$

Percebe-se que a potência instantânea é composta por um termo independente do tempo e um outro termo, uma onda cossenoidal.

2.3.2 Valor Eficaz

O Valor efetivo, ou *Root Mean Square (RMS)*, de uma onda periódica é determinado como valor médio quadrático da forma de onda. A equação 2.6 indica o valor eficaz de uma grandeza qualquer, $x(t)$ dentro de um período, T .

$$X_{RMS} = \sqrt{\frac{1}{T} \int_{t_0}^{t_0+T} [x(t)]^2 dt} \quad (2.6)$$

Veremos que o valor eficaz é extremamente útil para encontrar a potência média.

2.3.3 Potência Média

A potência média pode ser calculada integrando-se a potência instantânea ao longo de um período T , como apresentado pelas equações 2.7 e 2.8.

$$P_{mdia} = \frac{1}{T} \int_{t_0}^{t_0+T} p(t) dt \quad (2.7)$$

$$P_{mdia} = \frac{1}{2} V_m I_m \cos(\theta) \quad (2.8)$$

Com o uso dos valores *rms* de tensão, V_{rms} , e de corrente, I_{rms} , a Potência média é dada pela expressão da equação 2.9

$$P_{mdia} = V_{rms} I_{rms} \cos(\theta) \quad (2.9)$$

Na expressão, o termo composto por $V_{rms} I_{rms}$ é denominado potência aparente, S , e o termo $\cos(\theta)$ é denominado fator de potência.

2.3.4 Fator de Potência

O Fator de potência é a razão entre a potência média e a potência aparente, apresentado na equação 2.10.

$$fp = \frac{P_{mdia}}{P} = \cos(\theta) \quad (2.10)$$

O ângulo do fator de potência é diferença da fase da tensão e da fase da corrente de uma carga.

O fator de potência é uma grandeza elétrica que indica que uma carga absorve mais corrente que uma carga com fator de potência maior para realizar o mesmo trabalho. Ou seja, um sistema que apresente um fator de potência mais próximo de 1 será o sistema que absorverá menor corrente e conseqüentemente terá menos perdas.

2.3.5 Potência Complexa

Para declarar potência complexa é necessário utilizar as grandezas de tensão e corrente em sua forma fasorial. A notação fasorial de tensão e corrente estão apresentadas abaixo, respectivamente (NEPOMUCENO; COZENDEY, 2013).

$$\begin{aligned} \vec{V}_{rms} &= V e^{j\alpha} \\ \vec{I}_{rms} &= I e^{j\beta} \end{aligned}$$

A potência Complexa é, então, definida como o produto da tensão e do complexo conjugado da corrente, medida em VA.

$$\vec{S} = \vec{V}_{rms} * \vec{I}_{rms} \quad (2.11)$$

Por ser uma grandeza vetorial, a potência complexa apresenta uma parte real e uma imaginária, equação 2.12. Esta, também apresenta uma magnitude e fase, 2.13.

$$\vec{S} = P + jQ \quad (2.12)$$

$$\vec{S} = |\vec{S}|e^{j\phi} \quad (2.13)$$

A parte real da potência complexa é a potência ativa, P, sendo esta a potência que realiza trabalho na carga. A potencia ativa ou real é medida em Watts (W). A parte Imaginária é a potência reativa, Q, sendo esta referente ao armazenamento de energia, medida em var.

2.3.6 Energia elétrica

Lembrando que a potência varia com o tempo. Energia elétrica é a potência absorvida por uma carga ao longo do tempo, equação 2.14, e é medida em kWh.

$$E = \int_{t_0}^t P(t)dt \quad (2.14)$$

2.4 Aparelhos de medição

Segundo (NUNES, 2010), aparelhos de medição são instrumentos que proporcionam as grandezas medidas na forma de valores numéricos. Podendo estes serem fornecidos na forma digital ou analógica dependendo do instrumento utilizado.

2.4.1 Voltímetro

Os voltímetros são instrumentos utilizados para medir a tensão entre dois terminais de um circuito. Este deve ser ligado em paralelo com a carga para medir a diferença de potencial entre os terminais. A escala destes aparelhos normalmente é em Volts (V).

2.4.2 Amperímetro

Os amperímetros são instrumentos utilizados para medir a corrente entre dois terminais de um circuito. Deve ser ligado em série com a carga. As escalas destes aparelhos normalmente são em miliamperes (mA) ou amperes(A).

2.4.3 Wattímetro

O wattímetro é um instrumento que proporciona o produto da tensão e da corrente de um circuito, deste modo ele é uma associação de voltímetro e amperímetro. A medida da potência ativa é fornecida na escala de Watts (W) (CREDER, 2010).



Figura 2.3: Modelo de Wattímetro analógico(POLITERM, 2016)

2.4.4 Medidor de energia elétrica

O medidor de energia elétrica realiza a soma de potência absorvida por uma carga ao longo do tempo, deste modo, obtêm-se os valores de energia consumido em um determinado período. Este equipamento está presente nas residências e mensalmente a companhia de eletricidade realiza leituras dos valores destes medidores. A diferença nos valores entre uma leitura e outra é a quantidade de energia elétrica consumida pela residência durante aquele mês.



Figura 2.4: Modelo de medidor de energia elétrica residencial(SABERELETRICA, 2016)

2.5 Tempo real

Segundo (OLIVEIRA; FARINES; FRAGA, 2000), um sistema de tempo real "é um sistema computacional que deve reagir a estímulos oriundos do seu ambiente em prazos específicos". Isto é, a reação destes sistemas deve obedecer exigências dentro de um tempo determinado.

Aplicações envolvendo requisitos de tempo se tornam cada vez mais comuns na medida que os sistemas de automação predial e industrial se tornam mais complexos e extensos. A necessidade de manter estes sistemas vastos sob controle e de atender aos prazos de cada tarefa é o principal motivo da aplicabilidade de um sistema de tempo real em uma aplicação.

De acordo com (CARDOZO; OLIVEIRA; STRUMINSKI, 2017), a previsibilidade é o fator mais im-

portante na adoção de um sistema em tempo real, visto que este permite a identificação do comportamento do sistema e prevenção de falhas localizadas.

2.6 Precisão e calibração

Os instrumentos de medição sofrem com deterioração ao longo do tempo e com o uso. A partir disso erros sistêmicos passam a estar associados nas medições e podem interferir nos processos realizados pelo equipamento. A correção dos desvios que aparecerão nas medições garante a precisão das medições.

Visando analisar o desempenho de um equipamento para garantir que este forneça valores muito próximos ao medir o mesmo mensurado sob condições idênticas, deve-se realizar a calibração do equipamento.

Calibração é o processo que procura identificar, experimentalmente, o erro sistêmico associado ao equipamento e realizar a correção necessária para o funcionamento preciso do mesmo (ACC, 2015). Para realizar a calibração é necessário definir condições específicas às quais o equipamento a ser calibrado e o referencial estão estabelecidos. Através da comparação dos valores do referencial e dos valores do equipamento, determina-se o quanto o equipamento desvia dos valores da referência padronizada.

Após a identificação do erro associado ao equipamento é possível compensá-lo por meio de ajustes no equipamento ou através do *software* que faz o processamento das medições. Estes procedimentos aumentam a qualidade e confiabilidade dos equipamentos.

Capítulo 3

Metodologia

A fim de assegurar a funcionalidade do sistema proposto foi necessário elaborar uma estratégia para dividir o projeto em tarefas menores. Parte-se do módulo medidor de energia, figura 4.3, disponível no Laboratório de Automação e Robótica da UnB (LARA-UnB), elaborado como trabalho de conclusão de curso em 2013, (NEPOMUCENO; COZENDEY, 2013). O trabalho foi arquitetado com base neste módulo, deste modo, foi necessária uma análise das condições em que este se encontrava após um longo período sem uso.

Inicialmente, foi realizado um exame do módulo medidor existente para verificar o funcionamento dos componentes, fez-se a substituição de componentes queimados e reforço de soldas. Em seguida foi realizada uma análise da atuação do circuito de alimentação e dos sinais de entrada e saída da parte de medições.

Em seguida, foi implementada comunicação sem fio no módulo medidor. Sendo, também necessário adequar os códigos utilizados, anteriormente, para a implementação do módulo.

Dando continuidade, foi projetado um módulo medidor de temperatura que também servirá, para o propósito deste trabalho, como módulo receptor. Devido a disponibilidade limitada de um dos componentes que constituem os módulos, escolheu-se juntar o módulo receptor e o módulo de temperatura.

O sistema supervisório foi elaborado para aquisição e monitoramento dos dados de temperatura e parâmetros elétricos por meio do desenvolvimento de um banco de dados, no ScadaBR.

A realização de calibração dos módulos e de ensaios foram produzidos para verificação do funcionamento do sistema desenvolvido.

3.1 Calibração

O método utilizado para realizar a calibração dos instrumentos deste trabalho foi o comparativo, o qual é baseado na comparação da indicação medida e de um instrumento padrão adotado como referência.

Para realizar a calibração foram utilizados um wattímetro e um voltímetro disponíveis no LARA. O procedimento para calibração utilizado foi conectar o wattímetro em série com o módulo medidor de energia do sistema.

A partir dos resultados encontrados pelo wattímetro e pelo módulo será realizada uma comparação e será feita uma adequação do erro encontrado para correção do módulo. Cessando este procedimento teremos um módulo medidor seguindo o padrão de referência adotado.

3.2 Ambiente de estudo

Propõe-se a realização de testes experimentais no Laboratório de Robótica e Automação da Universidade de Brasília (LARA/UnB). O local do estudo é representado pela figura 3.1. A seção do laboratório em que serão realizados os testes é a sala representada pela figura 3.2.

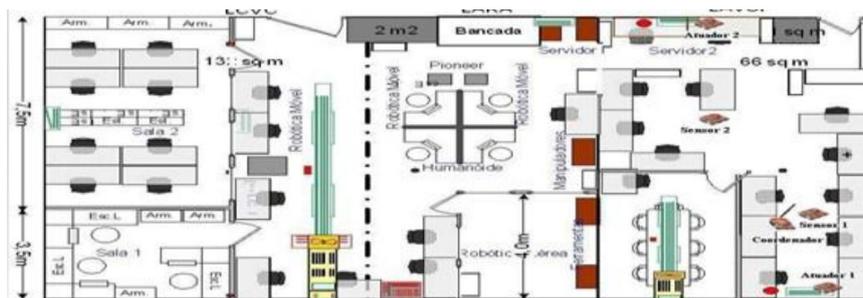


Figura 3.1: Laboratório de Automação e Robótica (LARA/UnB)

Este ambiente possui uma área de estudo comum que compreende uma área $6m^2$ e uma sala de reuniões que não faz parte do ambiente de estudo. A sala em análise contém 2 aparelhos de ar condicionado. Prevê-se a instalação dos módulos medidores nestes e a instalação do módulo de temperatura e receptor próximo ao centro da sala.

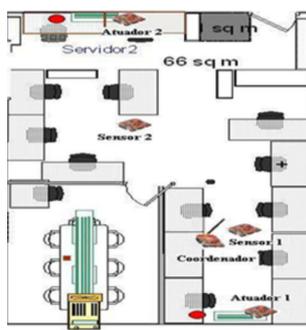


Figura 3.2: Sala do LARA/UnB

3.3 Equipamentos utilizados

Será utilizado para realizar os testes desejados um secador de cabelo da marca Remington da fabricante Microlite S.A., figura 3.3.

Este será utilizado como carga, por ser um equipamento que realiza troca de calor com o meio e ter elevada potência durante o funcionamento. Outra característica deste é a possibilidade de selecionar o



Figura 3.3: Secador de cabelo convencional

modo no qual este opera, sendo estes em média potência e em máxima potência. A facilidade de acionamento da carga e a possibilidade de alterar o modo de operação deste torna o secador de cabelo uma ótima opção para os testes. A partir dos resultados encontrados nos ensaios com o secador de cabelo pretende-se realizar um estudo para detectar alterações nos comportamento do aparelho a partir dos dados encontrados pelo sistema supervisorio.

As especificações técnicas do aparelho utilizado encontram-se na tabela 3.1.

Tabela 3.1: Especificações técnicas da carga utilizada

Parâmetro	Valor
Tensão	125/250V
Frequência	60/50Hz
Potência	1878 W
Modos de operação	2 Máx/med

O secador de cabelo foi utilizado por ser composto por um motor elétrico, o qual faz girar um rotor, criando um fluxo de ar. O aquecimento do ar é feito por uma resistência elétrica direcionada para a região de saída do ar. Tendo isto em vista espera-se encontrar um comportamento de natureza resistiva.

3.3.0.1 Instalação do medidor e acionamento das cargas

Por possuir 2 (dois) modos de operação, 110/220V, será escolhido o modo de operação em 220V por ser o nível de tensão na localidade do estudo.



Figura 3.4: modelo de instalação do módulo medidor

O medidor deverá ser conectado entre a carga e a rede elétrica. A figura 3.4 exhibe o posicionamento do medidor com relação ao secador, representado pelo *plug*, e a tomada.

3.4 Plano de ensaios adotado

Os ensaios serão feitos para o equipamento destacado anteriormente. Esperasse com estes ensaios obter curvas com os parâmetros elétricos adotando diferentes cenários.

Na busca do objetivo de sustentar o uso deste sistema na detecção de falhas em sistemas de ar condicionado, será adotada a seguinte estratégia. Analisar o comportamento de um secador de cabelo realizando alterações no comportamento deste para observar os efeitos causados nos parâmetros elétricos. A partir destes resultados com uma aparelho de aquecimento pode-se comparar com o comportamento esperado de equipamentos condicionadores de ar.

3.4.1 Ensaio de longa duração

Inicialmente, será realizado um teste com o medidor de energia com o secador de cabelo convencional, por algumas horas. Pretende-se validar a funcionalidade do sistema em longos períodos de tempo. Será utilizado um secador de cabelo como carga, por ser um equipamento que realiza troca de calor com o meio e ter fácil acesso a variação de potência utilizada..

O ensaio será realizado com o medidor conectado diretamente à entrada serial de um computador, ou seja, os dados serão transmitidos, nesta primeira fase, por um cabo entre o módulo e uma central.

Para o primeiro teste espera-se observar o comportamento do medidor por um período longo de tempo, de 1 (uma) a 2 (duas) horas, com acionamentos de carga ocasionais.

3.4.2 Ensaio com múltiplos acionamentos da carga

Por meio deste ensaio pretende-se identificar variações nos parâmetros medidos com a alternância entre os modos de operação da carga. O procedimento consistirá em realizar medidas variando o acionamento da carga entre seus modos de operação.

Neste ensaio será realizado com transmissão dos dados *wireless*.

Para o segundo teste espera-se observar o comportamento do medidor por um período menor com acionamentos de carga constantes nos dois modos de operação.

3.4.3 Carga com operação padrão

Neste ensaio espera-se obter um comportamento padrão para propósito de comparação com os demais ensaios. Inicialmente, será feito o acionamento da carga sem alterações para que os resultados encontrados sejam adotados como referência para comparação.

Os próximos ensaios visarão gerar alguma alteração no comportamento da carga de modo a simular alterações em sistemas de ar condicionado. Como mencionado anteriormente o resultado deste ensaio será utilizado para comparações com os resultados dos ensaios com alterações.

3.4.4 Ensaio com saída de ar parcialmente bloqueada

Neste ensaio a alteração a ser aplicada ao sistema será uma barreira física a ser colocada na saída de ar da carga. Mantendo a saída de ar do secador de cabelo parcialmente bloqueada durante todo o funcionamento da carga espera-se observar aumento na potência utilizada para operação deste. Em seguida, esperasse obter uma amostra para comparação entre os resultados em operação normal e com a barreira física instalada.

3.4.5 Ensaio saída de ar totalmente bloqueada

Este ensaio será iniciado com a saída de ar parcialmente bloqueada similar ao ensaio anterior, no entanto, durante o procedimento, a saída de ar deverá ser totalmente bloqueada para observação do comportamento. O objetivo deste teste será identificar a resposta do sistema com o superaquecimento.

3.5 Análise do erro experimental

O erro experimental está associado às medições na medida que existem incertezas envolvidas na confiabilidade dos equipamentos e nos resultados destes. Além dos erros relacionados aos equipamentos de medição, também devem ser considerados os erros aleatórios relacionados aos sistemas envolvidos (COLEMAN; STEELE, 2018). De acordo com (NEPOMUCENO; COZENDEY, 2013), os erros presentes estão associados à tolerâncias na malha de resistores e dos erros associados aos aparelhos de calibração.

Capítulo 4

Desenvolvimento

4.1 Visão geral do projeto

A existência de apenas um medidor de energia elétrica na entrada do fornecimento de um edifício impossibilita determinar o real consumo de cada equipamento, individualmente. Como discutido anteriormente, a forma de obter o monitoramento de equipamentos individualmente é pela instalação de um medidores exclusivos para cada um deles.

Assim sendo, o sistema explorado neste trabalho visa a instalação de um medidor para cada equipamento de ar condicionado ao qual se almeja monitorar. Para a realização de ensaios para detecção de falhas será utilizado um equipamento para ser monitorado e testar o funcionamento do sistema.

4.1.1 Sistema proposto

O sistema proposto consiste em fazer o monitoramento de uma carga, equipamento de ar condicionado, e da temperatura ambiente. Para cumprir este propósito, pretende-se a instalação de módulos capazes de medir energia elétrica e/ou temperatura num ambiente de estudo.

A figura 4.1 exibe um modelo do sistema proposto para determinação do consumo de um equipamento de refrigeração e da temperatura do ambiente. Como pode ser contemplado da figura, o sistema possui 3 (três) elementos, sendo eles um módulo responsável pela carga, um módulo encarregado de monitorar a temperatura e um módulo receptor das informações destes últimos.

O primeiro componente será um módulo que medirá os parâmetros de energia elétrica, o qual será instalado em equipamentos de ar condicionado. No caso de existirem mais aparelhos em um mesmo ambiente subentendesse a instalação de múltiplos módulos. Os parâmetros desejados deste medidor são tensão da rede, frequência, potência ativa e reativa.

O segundo componente será o módulo que avalia a temperatura do ambiente. Em ambientes grandes podem ser instalados mais de um módulo em diferentes localidades. No escopo deste trabalho prevê-se a instalação de uma unidade aproximadamente no centro do ambiente a ser estudado.

O último componente é o receptor, o qual estará conectado a um computador. A partir das informações

transmitidas pelos outros componentes do sistema, o receptor fará a integração destes com o sistema de monitoramento dos dados.

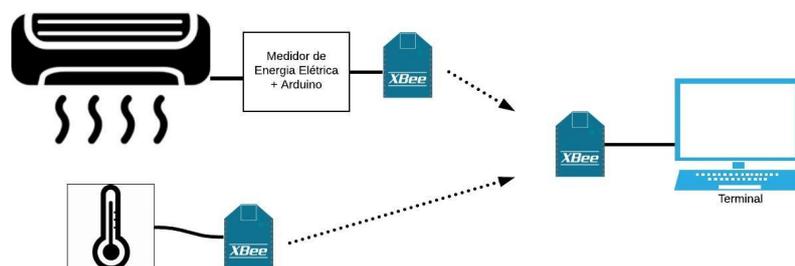


Figura 4.1: Modelo simplificado do sistema proposto

4.2 Módulo medidor de parâmetros elétricos

A estrutura completa do módulo está exibida na figura 4.2, a qual mostra a divisão em dois submódulos, sendo estes o de medição e o de transmissão. O módulo medidor partiu do circuito integrado (CI) SA9903B da Sames, o qual é encarregado por medir tensão, frequência da rede e potência ativa e reativa. Os componentes necessários para a composição da seção de medição tem como base o circuito proposto no *datasheet* do CI, (SAMES,). As outras seções constituintes do módulo são as responsáveis pelo processamento dos dados e pela transmissão dos mesmos, compostos pelo Arduino e XBee.

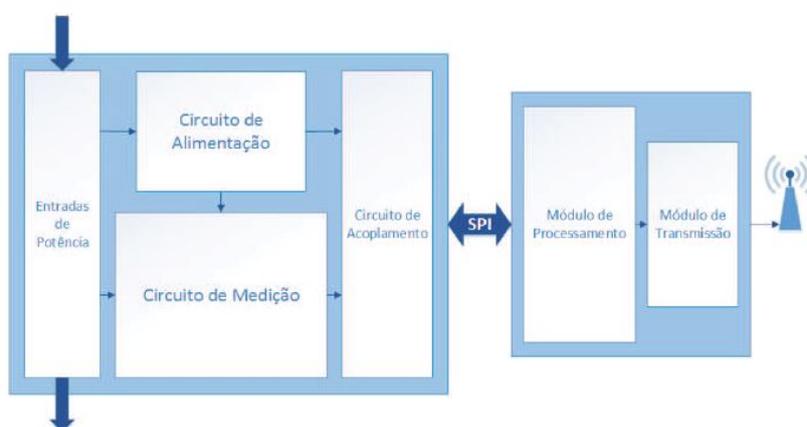
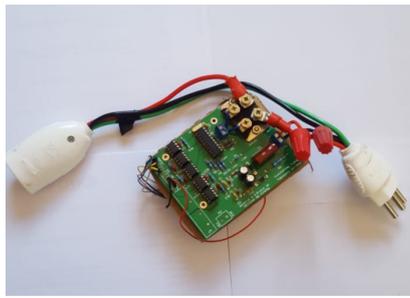


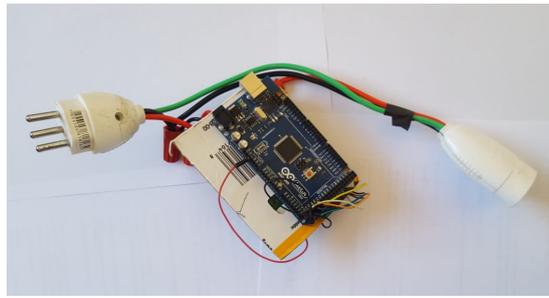
Figura 4.2: Estrutura do módulo
(NEPOMUCENO; COZENDEY, 2013)

4.2.1 Submódulo de medição

Como pode ser visto na figura 4.2, o submódulo de medição apresenta subdivisões. A composição de cada uma destas unidades será explicada a seguir.



(a) Parte frontal



(b) Parte traseira

Figura 4.3: Módulo medidor de energia utilizado no trabalho

4.2.1.1 Entrada de potência e circuito de alimentação

A entrada ao circuito de alimentação e de medição se dá diretamente da rede elétrica.

O circuito de alimentação é fundamentado no esquemático sugerido do *datasheet* do SA9903b, figura 4.4. A alimentação para o circuito de medição é uma fonte de -2,5V a 2,5V, que será aplicada no CI. O circuito de alimentação não interfere na medida do consumo de energia.

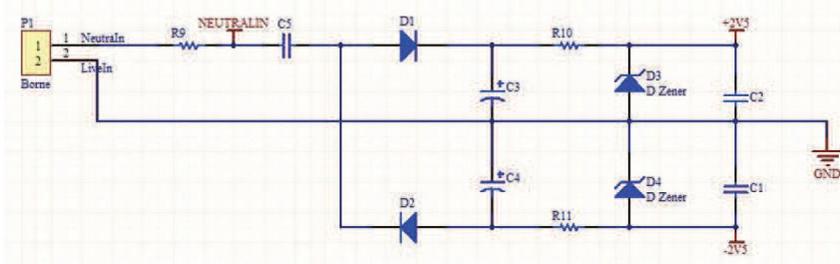


Figura 4.4: Circuito de alimentação (NEPOMUCENO; COZENDEY, 2013)

Durante a fase de investigação do funcionamento do módulo constatou-se que era necessário trocar o resistor de entrada. Esta alteração foi necessária para que o circuito entrasse de acordo com o indicado pelo *datasheet* (SAMES,).

4.2.1.2 Circuito de medição

A entrada da rede é conectada a um resistor *shunt*. Necessita-se medir a queda de tensão sobre o resistor e, então, obtém-se a corrente. Para encontrar a corrente o valor do resistor deve ser de alta precisão, deve ter baixa resistência para evitar perdas e não produzir uma queda de tensão grande que possa interferir no sistema que se quer medir. O modelo de resistor empregado foi o TGHGCR005FE-ND, novamente, adotado com base na queda de tensão requerido pelo CI da Sames para uma queda de tensão de 200mV. O resistor adotado possui uma resistência de 0.005Ω com tolerância de 1%.

O circuito responsável pelas medições de tensão, frequência, potência ativa e reativa tem como elemento principal o CI SA9903b. Este é um medidor de potência monofásico que foi adotado, de acordo com (NEPOMUCENO; COZENDEY, 2013), considerando requisitos de projetos como tensão máxima da

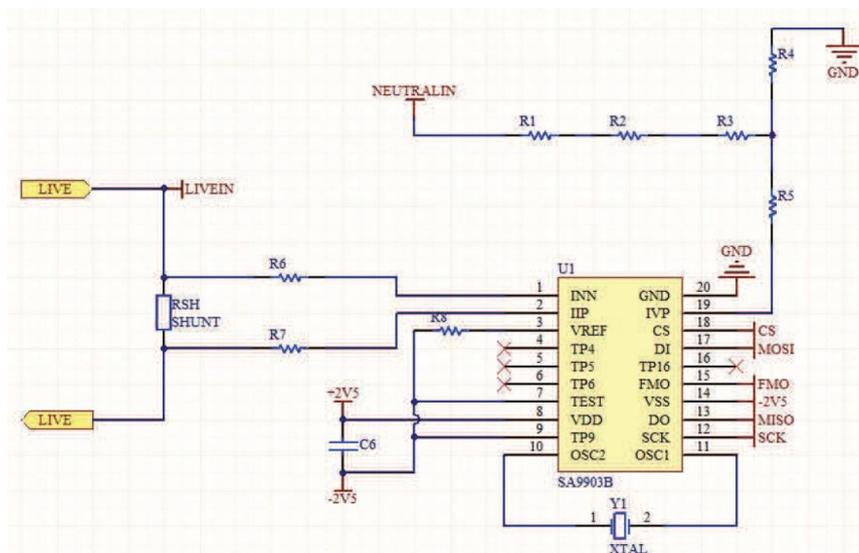


Figura 4.5: Circuito de medição de tensão, corrente e potência, utilizando um SA9903B (NEPOMUCENO; COZENDEY, 2013)

rede de 230V e corrente máxima de 40A. Por se tratarem de equipamentos com elevadas potências, os aparelhos de ar condicionado e outros equipamentos de troca de calor requerem uma corrente elevada. Por este motivo os requisitos do projeto consideraram correntes com valores elevados, compreendendo um total de 9,2 kW de potência máxima suportada. O diagrama do circuito de medição de tensão, corrente e potência está ilustrado na figura 4.5

A corrente fornecida pelo resistor *shunt* é inserida nas entradas *Current Sense Input* (IIP e IIN) do CI e a tensão é inserida em *Voltage Sense Input* (IVP), sendo necessário aplicar um divisor de tensão neste. Mais detalhes da configuração do circuito estão disponíveis no anexo I e no datasheet do SA9903b.

O sinais de saída do CI que importam para a obtenção das medidas são o *Serial Data Out* (SDO) e o *Mains Voltage Sense Zero* (FMO). Estes são a saída serial de dados e sinal seguidor de passagem por zero, respectivamente. Estes são enviados via barramento SPI para o submódulo de transmissão. "A interface SPI é um barramento síncrono utilizado na transferência de dados entre o controlador e o SA9903B. Os pinos DO (*Serial Data Out*), DI (*SERIAL DATA IN*), CS (*Chip Select*) e SCK (*Serial Clock*) são utilizados na execução do barramento. Neste caso, o SA9903B é configurado como dispositivo escravo (*slave*) e o controlador é configurado como dispositivo mestre (*master*). A entrada CS inicia e termina a transferência de dados. O sinal SCK (gerado pelo controlador) sincroniza os sinais a leitura e escrita de dados. DO e DI são, respectivamente, a saída e entrada de dados do SA9903B." de (NEPOMUCENO; COZENDEY, 2013).

4.2.1.3 circuito de acoplamento

Apesar de já possuir os sinais/registadores que contém os dados desejado de tensão, frequência e potência, estes sinais não podem ser enviados diretamente para o submódulo de transmissão. A justificativa é que estes operam na faixa de tensão de -2,5 V a +2,5 V enquanto o dispositivo Arduino opera entre 0 V e 5,0 V. É necessário então fazer uma conversão destes níveis lógicos para assegurar a leitura do SPI. Para isto, foram utilizados dispositivos optoacopladores. Dispositivos eletrônicos que transmitem sinais

elétricos por meio de ondas eletromagnéticas, utilizando um LED que transmite os sinais opticamente acoplado a um fotodetector garantindo o isolamento físico e elétrico entre os sinais de entrada e saída. Sua principal função é fazer a tradução das tensões e evitar que sinais de alta tensão, conectados à rede elétrica, danifiquem os circuitos do submódulo de transmissão. A figura 4.6 exibe o esquemático do acoplamento adotado.

Após a tradução teremos as saídas do circuito de medição, sendo estas: *Serial Clock (SCK)*, *Chip Select (CS)*, *FMO*, *Master Input Slave Output (MISO)* and *Master Output Slave Input (MOSI)*. As medidas adquiridas pelo SA9903b são armazenadas em registradores de 24 bits e usa-se um barramento SPI (*Serial Peripheral Interface*) para envio destes para o arduino.

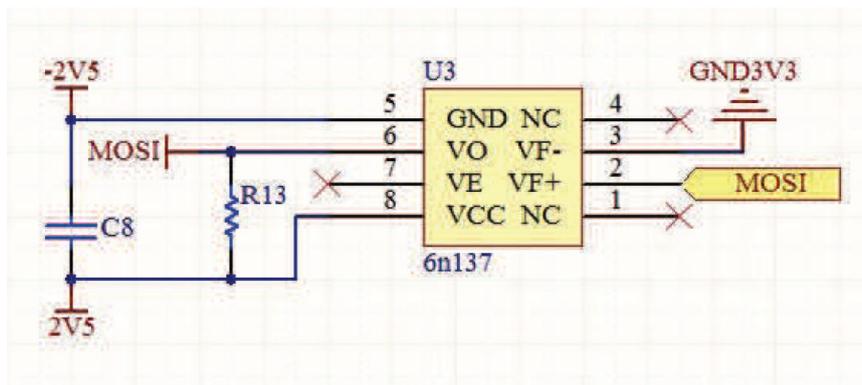


Figura 4.6: Ligação entre a saída de dados medidos com a entrada do processamento (NEPOMUCENO; COZENDEY, 2013)

4.2.2 Arduino

Arduino é uma plataforma *open-source* de hardware e software que busca facilitar o uso destes por meio do *software* de Arduino (IDE) o qual utiliza a linguagem de programação de arduino. A linguagem de programação usada na plataforma do arduino é bastante próxima da C++. As placas de arduino, que tem como componente principal um microcontrolador, são utilizadas em muitas aplicações desde projetos do dia a dia à mais sofisticados instrumentos científicos. Justamente por ser muito utilizado existe uma vasta quantidade de informações e bibliotecas disponibilizadas pela comunidade que utiliza a plataforma.

Todas as placas de Arduino são completamente *open-source* permitindo a montagem destas pelo usuário ou a compra destas no mercado a preços mais acessíveis comparadas com outras plataformas de microcontroladores.

O arduino foi escolhido para ser utilizado nos módulos atuais por ter tamanho reduzido e por satisfazer a todas necessidades para o desenvolvimento dos módulos. A placa de arduino MEGA 2560 apresenta portas de alimentação, portas digitais e analógicas. As entradas digitais só podem assumir dois estados, alto (*HIGH*) ou baixo (*LOW*). A escolha das portas para a realização das funções deve levar em conta a natureza do sinal a ser trabalhado pelo arduino. A figura 4.7 exibe a placa de arduino implementada juntamente com o circuito de medição do módulo.

Durante a fase de estudo do projeto, buscamos entender como funcionava a programação de cada seção



Figura 4.7: Modelo simplificado do sistema proposto

e componente a ser utilizado no módulo. O código utilizado para implementar o módulo de medição no arduino está disponível nos anexos.

4.2.3 Circuito comunicação

A transmissão de dados neste trabalho foi feita tanto serialmente quanto com o uso de um módulo de radiofrequência Xbee.

XBee é desenvolvido e distribuído pela Digi International e tem como característica o baixo custo para realizar uma comunicação sem fio por radiofrequência. O xbee foi selecionado para o projeto por estar disponível no Laboratório e por ser compatível com o arduino através do uso de um *shield* para acoplamento deste a placa de arduino. A versão de XBee disponível é a 802.15.4 Series 1, figura 4.8.



Figura 4.8: Módulo XBee

4.2.4 Logica de programação

A configuração do *software* aplicado necessita da biblioteca MODBUS, a qual é responsável pela função de leitura de registradores do tipo *holding* e envio destes para o supervisor. Além desta biblioteca foi utilizada a biblioteca SA9903b que é responsável pela leitura dos registradores do CI sa9903b e realiza as devidas correções que se façam necessárias para identificar os parâmetros de tensão, frequência de rede, potência ativa e reativa. O protocolo MODBUS foi empregado por permitir a comunicação de múltiplos dispositivos numa mesma rede. O módulo medidor de energia foi projetado para fazer o envio de dados tanto por fio quanto sem fio, ambos serialmente, portanto, a programação precisou levar em conta estas duas possibilidades para o módulo.

O programa consiste em adquirir os dados dos registradores do CI e são encaminhados para o modbus, o qual os enviará para o supervisório.

4.3 Módulo medidor de temperatura

Para conseguir realizar o monitoramento da temperatura do ambiente foi elaborado um módulo medidor de temperatura com o fim de identificar a temperatura e informar ao supervisório. O módulo de medição de temperatura é constituído de um sensor de temperatura LM36, um Arduino Mega 2560 e um Xbee.

4.3.1 Sensor de temperatura

O sensor de temperatura utilizado é um *Linker Thermal Module*, figura 4.9, da *LinkSprite* (LINKERSPITE,). Este é formado por um circuito integrado de temperatura LM35, o qual faz a detecção da temperatura do ambiente e informa uma saída em escala de tensão linearmente proporcional a escala centígrada.

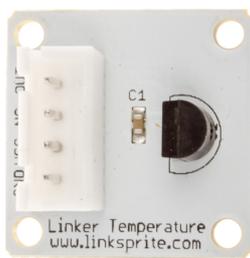


Figura 4.9: Sensor de Temperatura
Fonte:(LINKERSPITE,)

De acordo com o fabricante (TMP35,), o LM35 não precisa de uma calibração externa, apresentando uma acurácia de $\pm 1/4^{\circ}\text{C}$ na temperatura ambiente (25°C). Este opera em uma escala de -40°C a 150°C . A relação linear se dá na forma de 0,1V na saída representar a temperatura de -40°C , a medida que aumenta-se 1°C a tensão sofre um acréscimo de 10mV. Desta forma é possível calcular a temperatura que está sendo medida com base na equação 4.1.

$$V_{out}^{\rightarrow} = 10 \frac{mV}{^{\circ}C} \tau \quad (4.1)$$

em que,

- V_{out} é a tensão de saída medida pelo sensor, em V;
- τ é a temperatura em $^{\circ}\text{C}$;

4.3.2 Processamento do módulo

O arduino foi utilizado para a associação do sensor de temperatura, *Linker Thermal Module*, com a comunicação para o supervisor. Deste modo, o arduino é responsável por adquirir os dados de tensão do sensor LM35 e pelo envi destes para o ponto de comunicação com a central supervisória. A figura 4.10 indica o módulo desenvolvido para o monitoramento de temperatura.

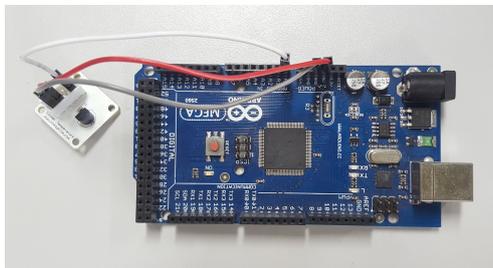


Figura 4.10: Módulo de medição da temperatura

4.3.2.1 Xbee

Juntamente com o arduino foi implementado um Xbee com o objetivo de proporcionar comunicação sem fio entre o módulo e a central do supervisor.

4.3.2.2 Logica de programação

A configuração do *software* aplicado usa a biblioteca biblioteca MODBUS, a qual é responsável pela função de leitura de registradores do tipo *holding*. O protocolo MODBUS foi empregado por permitir a comunicação de múltiplos dispositivos numa mesma rede. O módulo medidor de temperatura foi projetado juntamente com o módulo receptor. Desta forma a programação faz o envio de dados por fio com o computador central da rede. A estrutura do programa utilizado para implementação do módulo de temperatura está exposta na figura 4.11.

A lógica empregada começa pela identificação da porta da placa arduino que recebe o sinal do sensor. Em seguida, o programa efetua a leitura deste sinal de tensão da porta, indicada anteriormente, e geram dois registradores. Estes são os registradores para a o valor da temperatura, em °C, e da dimensão do registrador *holding*.

Subsequentemente, realiza-se a conversão do valor de tensão lido na porta de entrada para a escala de tensão de 5V, que corresponde a alimentação do arduino. Após a obtenção do valor corrigido de tensão, divide-se este por 1024.

O próximo passo é realizar a conversão da tensão corrigida para temperatura em graus Celsius. A conversão leva em conta a equação 4.1, segundo a qual indica que a relação linear se dá na forma de 10mV de acréscimo na tensão aumentou-se em 1°C a temperatura. Em tracejado na figura 4.11 está apontada a possibilidade de realizar a conversão para a escala de graus fahrenheit caso fosse desejado.

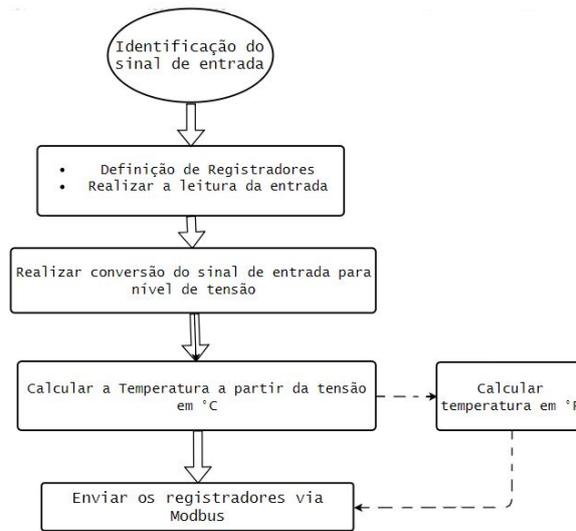


Figura 4.11: Estrutura do programa do medidor da temperatura

Por último, atribui-se o valor da temperatura calculado para o registrador correspondente para, posteriormente, ser lido pelo supervisório. O programa repete esse ciclo registrando todos os valores de temperatura com um atraso de 1(um) segundo entre cada valor registrado. Código disponível no anexo III.1

4.4 SCADA

SCADA (*supervisory control and data acquisition*), ou supervisório para controle e aquisição de dado, é um sistema capaz de controlar e supervisionar processos em larga escala. De acordo com (BOYER, 2018), SCADA é uma tecnologia que permite a coleta de dados de um grupo de equipamentos ou instalações e permite o envio de comandos para estes de forma controlada. Este sistema possibilita que o usuário possa operar todo o conjunto instalado remotamente.

Os elementos de um sistema SCADA são uma central do operador, RTUs (*remote terminal unit*), atuadores, sistema de comunicação, instrumentos, sensores e outros dispositivos para controle. A partir de cada RTU são obtidos os dados, os quais são enviados à central pelo mecanismo de comunicação adotado. Neste trabalho foram utilizados o módulo de medição de temperatura e parâmetros elétricos como RTUs e um módulo central. A comunicação adotou um Xbee como mencionado em 4.2.3.

Neste trabalho foi adotado o sistema supervisório *open source* ScadaBR. A escolha deste foi feita por ser um sistema gratuito e utilizado em automação predial (SCADABR, 2014). A figura 4.12 exibe a interface do ScadaBR acessado pelo navegador de internet.

O primeiro procedimento adotado para a implementação de um sistema supervisório utilizando o ScadaBR foi criar um *data source* que coletará os dados enviados via protocolo modbus para o supervisório. A

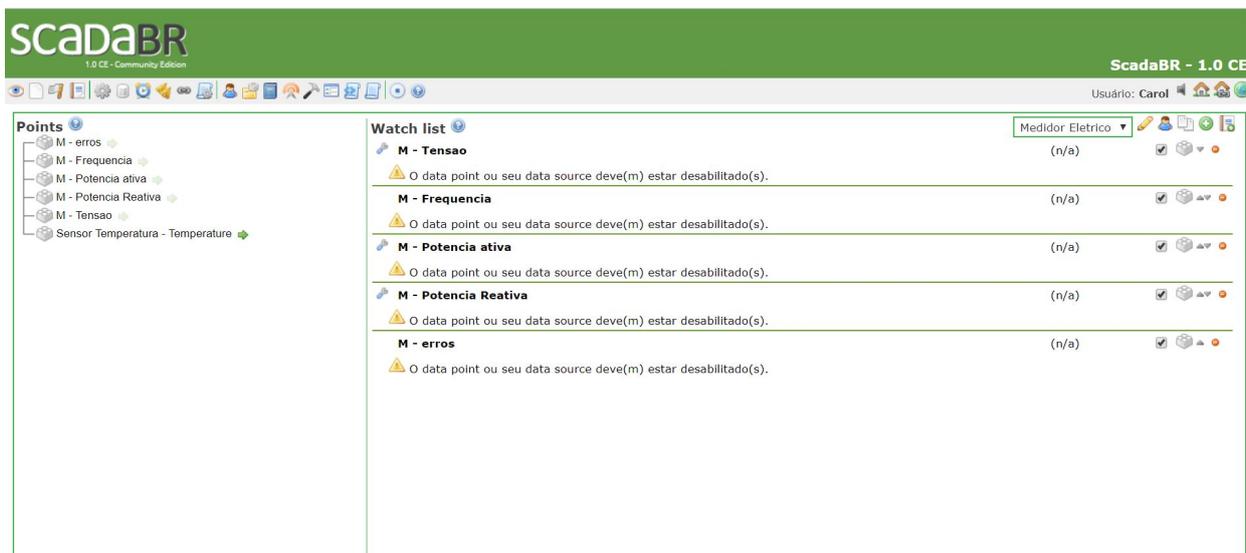


Figura 4.12: Interface do sistema ScadaBR

figura 4.13 mostra os dois *data sources* implementados no âmbito deste trabalho. Sendo estes responsáveis pelas requisições de leitura das RTUs do módulo medidor de energia e do módulo medidor de temperatura.

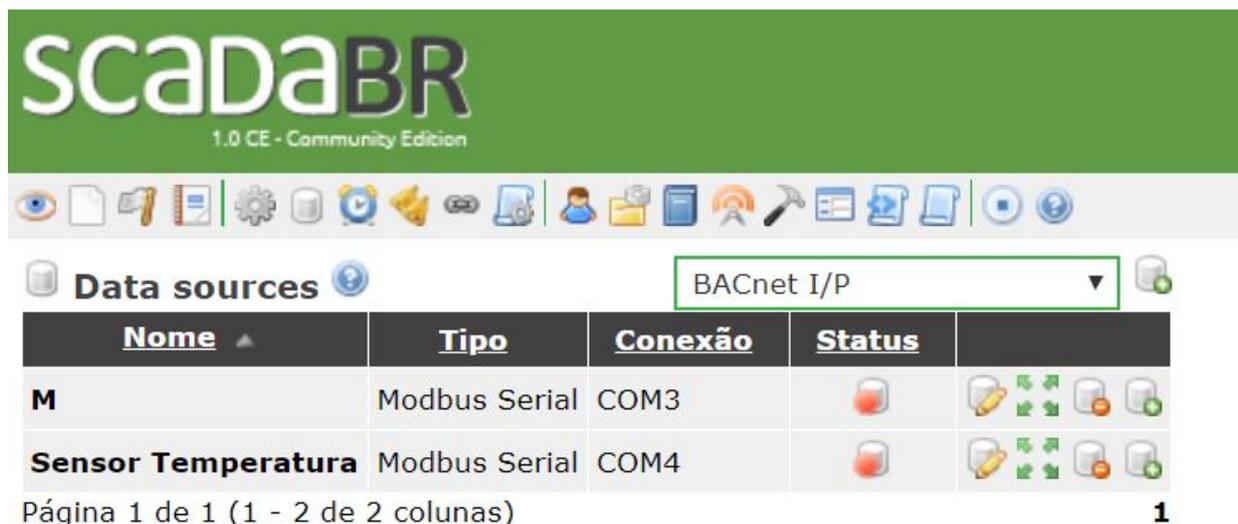


Figura 4.13: *data sources* criados para o supervisão desenvolvido

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
erros	Numérico		1	Registrador holding 0	
Frequencia	Numérico		1	Registrador holding 2	
Potencia ativa	Numérico		1	Registrador holding 3	
Potencia Reativa	Numérico		1	Registrador holding 4	
Tensao	Numérico		1	Registrador holding 1	

Figura 4.14: *data points*

A cada 1000 ms o modbus realiza a leitura dos registradores processados pelo arduino. Estas devem ser identificadas separadamente em *data points*, figura 4.14. Estes armazenam todas as leituras recebidas referentes a cada grandeza medida e tratada pelos registradores. Como cada *data point* representa uma grandeza a separação destes levou em conta as grandezas que são adquiridas pelo módulo medidor. Portanto, foram criados os *data points* de tensão, frequência da rede, potência ativa e reativa, além de um *data point* para os erros de transmissão. A figura ?? Para a RTU do módulo de temperatura foi criado apenas um *data point*, destinado à temperatura exibido na figura 4.16.

The screenshot displays a software interface for configuring a Modbus RTU connection. It includes several panels:

- Alarmes vigentes:** A status bar indicating no active alarms for the data source.
- Propriedades do modbus serial:** Configuration for the serial connection, including Name (M), Export ID (XID) (DS_126078), update period (1 second), timeout (500 ms), and attempts (2). It also shows baud rate (57600) and other communication parameters.
- Pesquisa de nós modbus:** A search panel for finding Modbus nodes, currently showing no results.
- Leitura de dados modbus:** Configuration for reading data from a slave (ID 1), including register range (Status do coil) and number of registers (100).
- Teste de localizador de ponto:** Configuration for testing a point locator, including slave ID (1), register range (Status do coil), and data type (Binário).
- Níveis de alarme de eventos:** Configuration for event alarm levels, currently set to 'Nenhum alarme'.
- Data points:** A table listing configured data points:

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
erros	Numérico		1	Registrador holding 0	0
Frequencia	Numérico		1	Registrador holding 2	2
Potencia ativa	Numérico		1	Registrador holding 3	3
Potencia Reativa	Numérico		1	Registrador holding 4	4
Tensao	Numérico		1	Registrador holding 1	1
- Detalhes do data point:** Configuration for a specific data point named 'erros', including its Export ID (XID) (DP_554119), slave ID (1), register range (Registrador holding), and data type (Inteiro de 2 bytes sem sinal).

Figura 4.15: Detalhes de um *data point*

O monitoramento dos dados adquiridos durante os ensaios é feito por meio de uma *watch list* que apresenta as grandezas dos *data points* a medida que as leituras são adquiridas pelo supervisor. As representações destas variáveis também aparecem através da interface de cada *data point*.

Neste trabalho, o MATLAB foi utilizado para a construção dos gráficos a partir dos dados dos parâmetros de energia elétrica e de temperatura. Importa-se os dados do ScadaBR, para o MATLAB, a da geração de um arquivo .csv no ScadaB. A partir disto é possível traçar os gráficos de tensão RMS, frequência da rede, potência ativa e reativa. Além disso, calculou-se a magnitude da potência aparente, equação 2.13, e o

Data points					
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Temperature	Numérico		1	Registrador holding	0

Detalhes do data point	
Nome	Temperature
Export ID (XID)	DP_061215
Id do escravo	1
Faixa do registro	Registrador holding
Tipo de dados modbus	Inteiro de 2 bytes com sinal
Offset (baseado em 0)	0
Bit	0
Número de registradores	0
Codificação de caracteres	ASCII
Configurável	
Multiplicador	0.262307692
Aditivo	-33.64615385

Figura 4.16: Estrutura do medidor da temperatura

fator de potência, equação 2.10.

Capítulo 5

Resultados Experimentais

Seguindo a metodologia indicada no capítulo 3 os seguintes resultados foram obtidos. Esperava-se encontrar as curvas de frequência da rede, de tensão, de potência ativa, reativa, potência aparente e fator de potência dos ensaios realizados. No fim, há um comparativo que demonstra a relevância do trabalho.

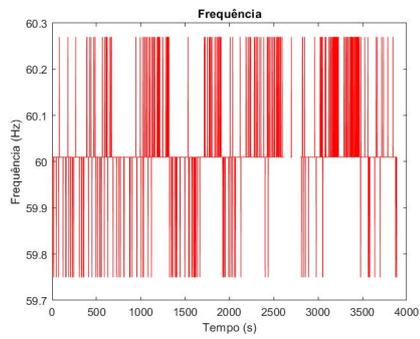
5.1 Calibração e ajustes

O primeiro teste visando a calibração do módulo medidor de energia elétrica consistiu em verificar a medição do nível de tensão. O voltímetro fazia uma leitura de 209V enquanto o módulo nos retornava 205V. Este desvio foi considerado satisfatório por desviar em menos de 2% do voltímetro adotado como referência. Em seguida, procurou-se verificar a potência ativa medida pelo módulo. Com o uso do secador de cabelo, o wattímetro mediu 1076W enquanto o módulo registrou 975W, sendo um desvio de aproximadamente 9,5%. Estes resultados foram considerados aceitáveis por não desviarem em mais de 10% para a realização deste trabalho. Deve-se levar em conta a precisão quanto à calibração do voltímetro e wattímetro adotados como referência.

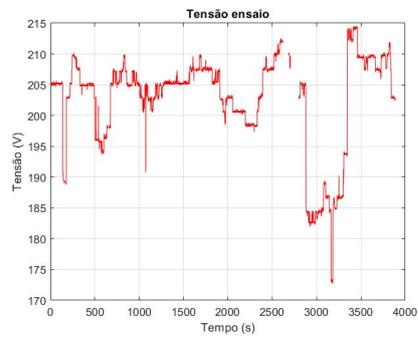
5.2 Ensaio de longa duração com módulo medidor de energia

O primeiro teste com o módulo medidor de energia foi realizado utilizando o secador de cabelo convencional. A figura 5.1 apresenta os gráficos obtidos a partir das medições realizadas dentro de um período de aproximadamente três horas com um acionamento do aparelho secador de cabelo por alguns minutos. Durante este período ocorreu um acionamento da carga por cerca de 3 minutos, pode-se ver que, nas figuras 5.1(c), 5.1(e) e 5.1(d), ocorre um pico da potência medida indicando o acionamento da carga, porém não é possível ver alterações na frequência e no fator de potência devido aos gráficos representarem um tempo muito longo.

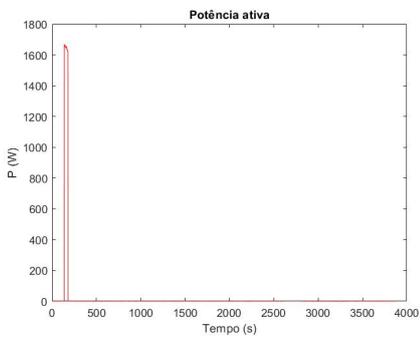
Outro ponto a se observar é a variação de tensão, figura 5.1(b), que ocorreu neste período analisado. Como o acionamento da carga ocorreu nos primeiros minutos do período estudado, as variações observadas na tensão devem-se a variações na rede, uma vez que a carga encontrava-se desligada.



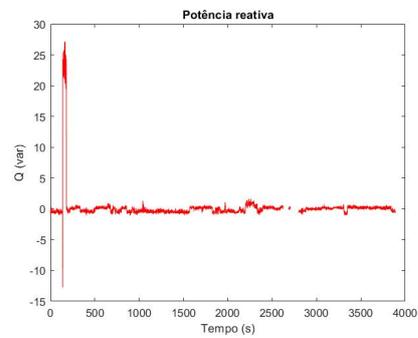
(a) Frequência



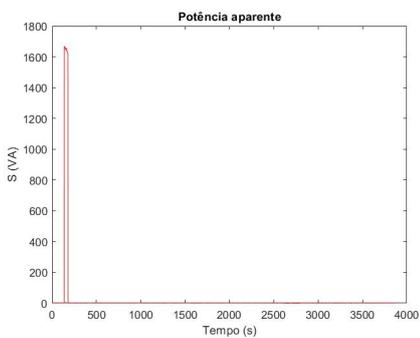
(b) Tensão



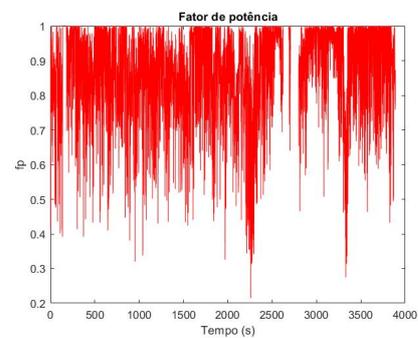
(c) Potência Ativa



(d) Potência Reativa



(e) Potência Aparente



(f) Fator de potência

Figura 5.1: Gráficos dos parâmetros elétricos ao longo do período de ensaio de longa duração

5.3 Ensaio com múltiplos acionamentos da carga em diferentes modos de operação

Neste ensaio procurou-se acionar a carga em seus dois modos de operação e monitorar a variação dos parâmetros com o módulo medidor de energia a medida que as alterações ocorriam. A figura 5.2 apresenta os gráficos obtidos a partir das medições realizadas dentro de um período menor desta vez. Por 7 minutos o medidor esteve conectado à carga.

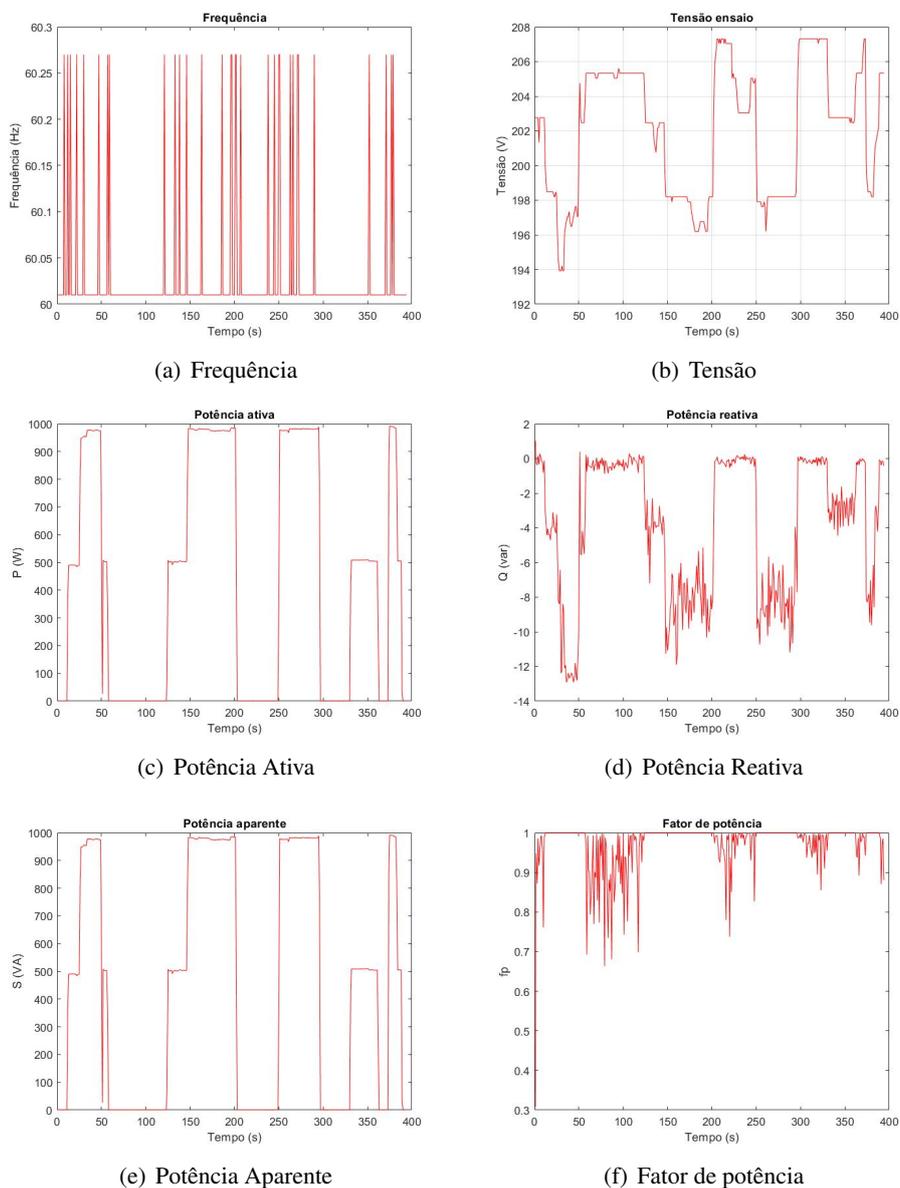


Figura 5.2: Gráficos dos parâmetros elétricos ao longo do ensaio de acionamentos alternados

Durante este período ocorreram acionamentos alternados da carga, tanto no modo de mínima potência quanto no de máxima potência. Nas figuras 5.6(a), 5.6(b) e 5.6(c), pode-se observar os momentos em que ocorreram os acionamentos do secador de cabelo. Percebem-se 2 faixas em que o secador opera, sendo estas em média potência e em máxima potência. Na operação no modo de média potência, a potência

utilizada para o funcionamento do secador medida foi em torno de 500W. No modo de máxima operação, a potência medida foi na faixa de 980W.

5.4 Testes para detecção de falhas

Na busca de viabilizar o uso deste sistema na detecção de falhas em sistemas de ar condicionado, foi adotada a estratégia de analisar o comportamento de um secador de cabelo e realizar alterações no comportamento deste para observar os efeitos causados nos parâmetros elétricos.

5.4.1 Carga com operação padrão

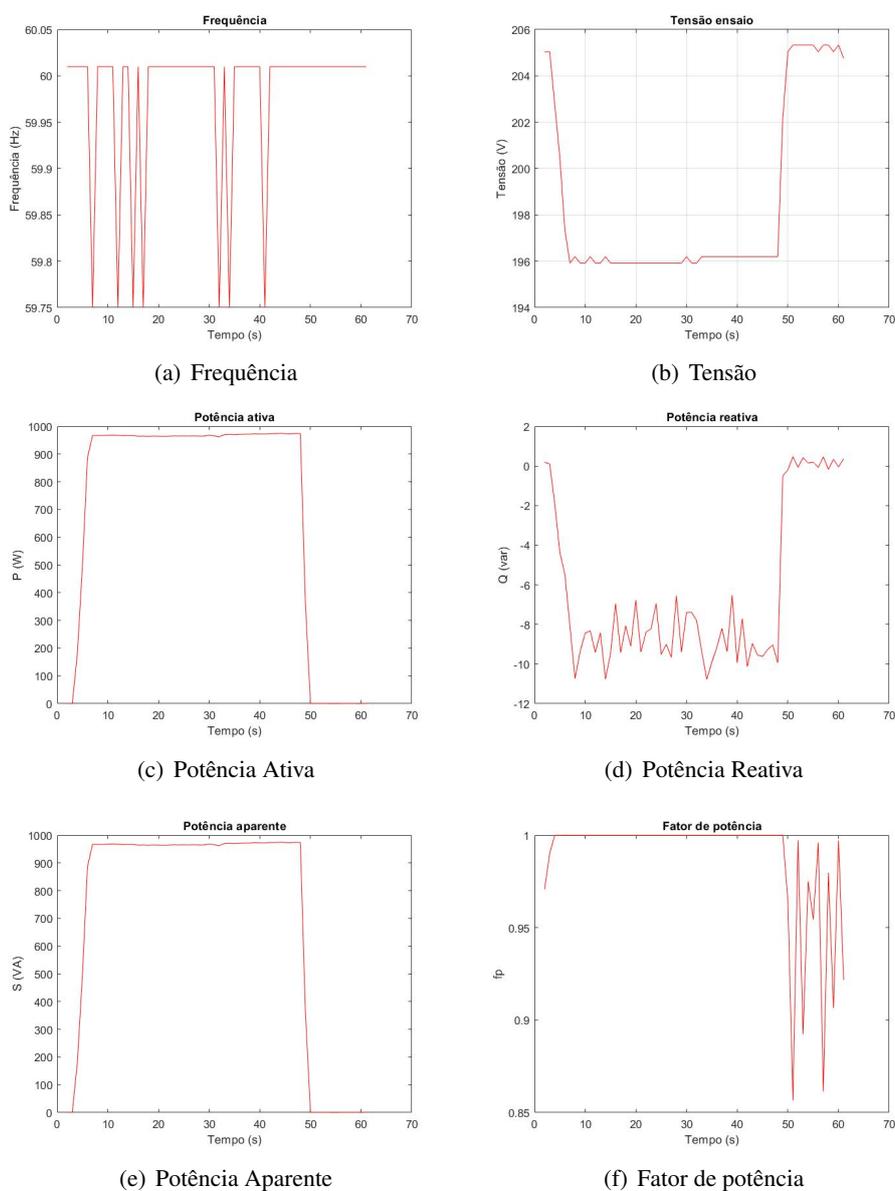


Figura 5.3: Gráficos dos parâmetros elétricos ao longo do período do ensaio padrão

Inicialmente procurou-se analisar o funcionamento da carga sem alterações para ser adotado como um padrão de comparação nos próximos ensaios. A figura 5.3 exibe os resultados encontrados no ensaio de operação padrão do secador de cabelo acionado no modo de operação de máxima potência.

Como mencionado anteriormente este resultado será utilizado para comparações com os próximos resultados encontrados. Pode-se observar que a máxima potência ativa registrada foi de 974.68 W e em média -8.64 var durante a operação do secador. Na tensão não foram observadas muitas variações durante o funcionamento do motor.

5.4.2 Ensaio com saída de ar parcialmente bloqueada

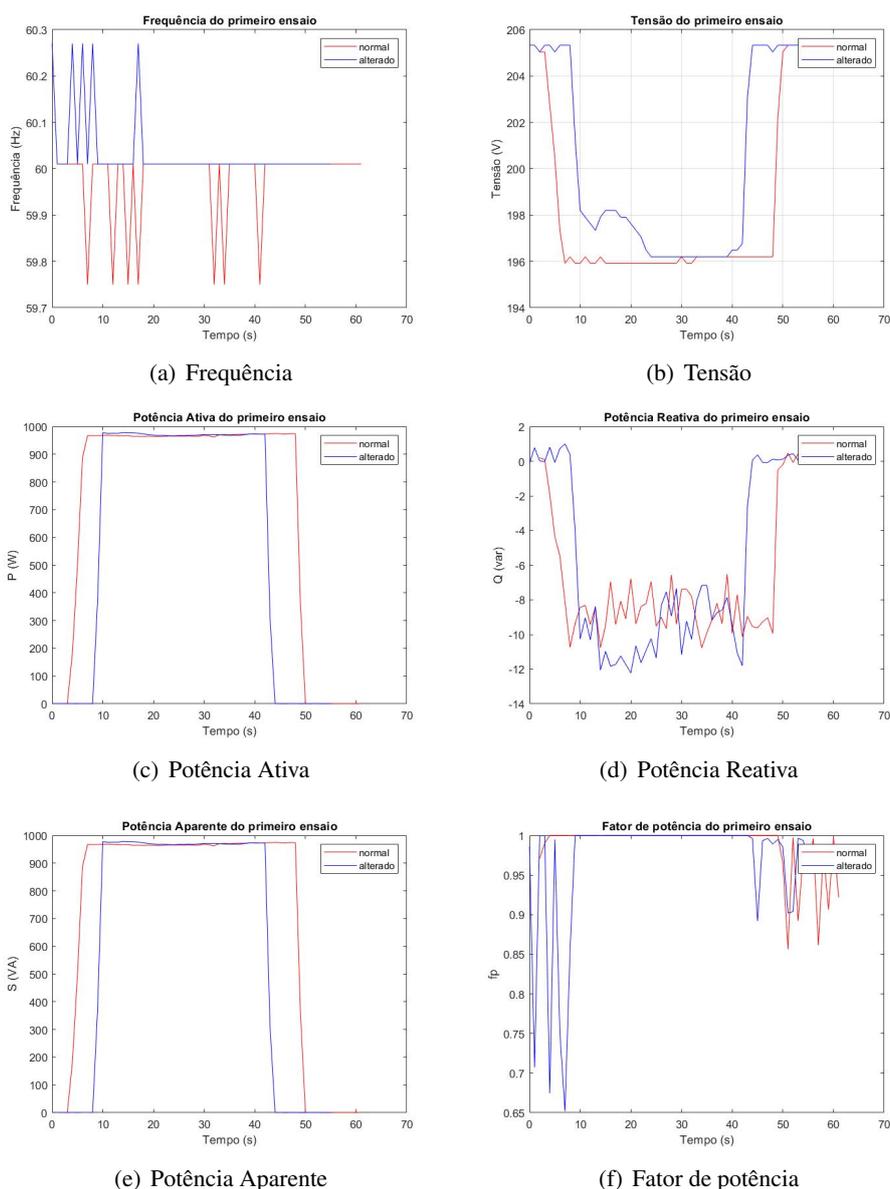


Figura 5.4: Gráficos comparativo dos parâmetros elétricos no ensaio com saída de ar parcialmente bloqueada

Neste ensaio a alteração aplicada ao sistema foi manter a saída de ar do secador de cabelo parcialmente

bloqueada durante todo o funcionamento da carga. A figura 5.4 exibe o comparativo entre os resultados encontrados no ensaio com o ensaio padrão realizado anteriormente.

A partir dos gráficos comparativos entre a operação padrão e com o secador com saída de ar parcialmente bloqueada, pode-se observar que ocorreu um aumento pequeno na potência registrada durante o funcionamento da carga. A alteração foi registrada na figura 5.4(c) e 5.4(d). A maior variação que pode ser percebida ocorre imediatamente após o acionamento da carga.

5.4.3 Ensaio secador de cabelo com saída de ar totalmente fechada durante funcionamento

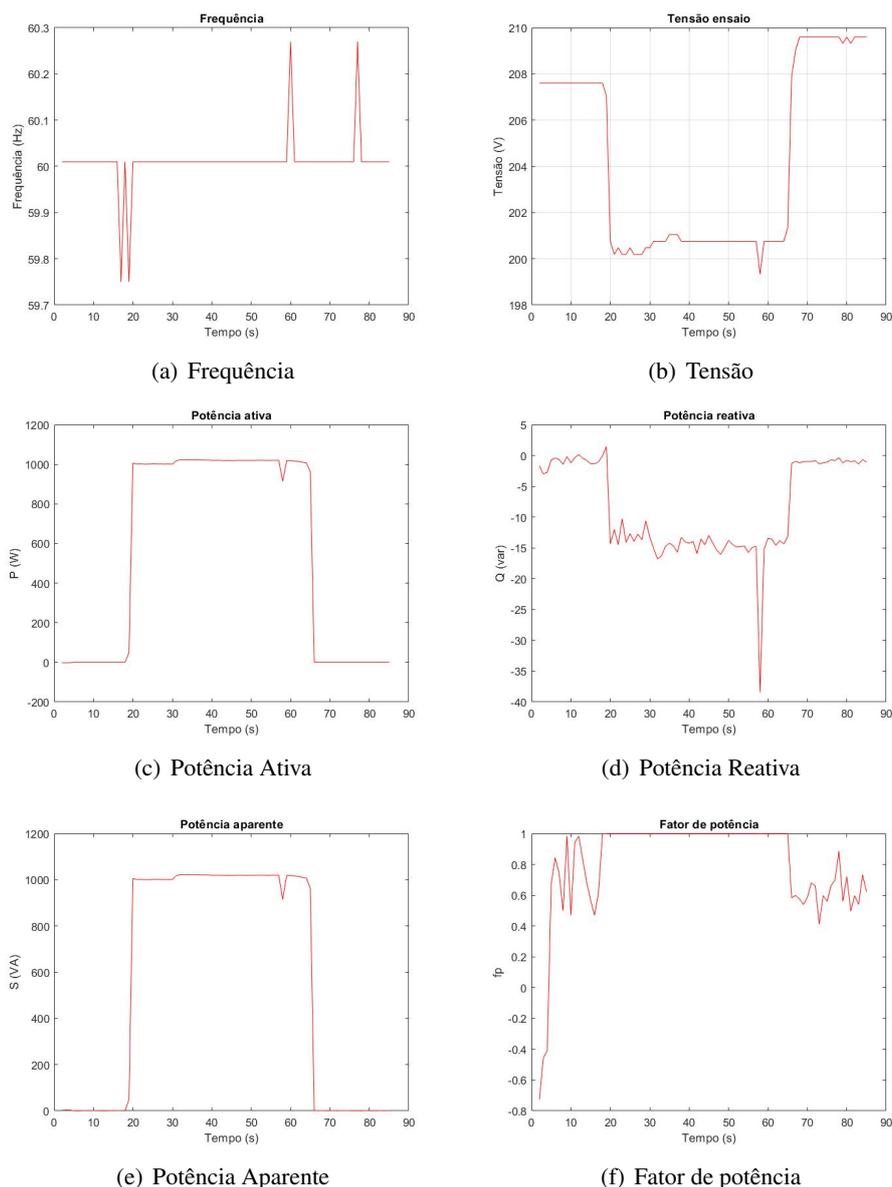


Figura 5.5: Gráficos dos parâmetros elétricos do ensaio com saída de ar obstruída

Este ensaio foi iniciado com a saída de ar parcialmente bloqueada similar ao ensaio anterior, no entanto, durante o procedimento, o saída de ar foi totalmente bloqueada. O objetivo deste teste era identificar a resposta do sistema a entrada de uma falha de aplicação real. Como se observa a figura 5.5 apresenta

os resultados encontrados durante o teste. O momento em que a saída de ar é totalmente bloqueada é identificada pelas alterações após 57 segundos após o início do ensaio.

A figura 5.6 apresenta o comparativo do ensaio padrão com o ensaio de bloqueio da saída de ar. Observa-se que a medida de consumo é 0 W com o secador desligado e em torno de 974,68 W na potência máxima durante o funcionamento normal e de 1022,3W no funcionamento com a saída parcialmente obstruída. Além disso observa-se uma queda no momento de obstrução total de 10.5% da potência para funcionamento do motor. Em seguida, o secador voltou a demonstrar comportamento que adotava com a saída parcialmente. Após a obstrução total da saída de ar ocorreu superaquecimento do secador e este desligou após 10 segundos.

Nota-se que as variações de potência reativa tem correspondência simultânea das variações de potência ativa. Pela natureza resistiva da carga percebe-se que a potência ativa predomina no consumo total. Pela ordem de grandeza da potência reativa comparada com a da ordem de grandeza de potência ativa, percebe-se uma aglomeração em níveis baixos de potências reativas.

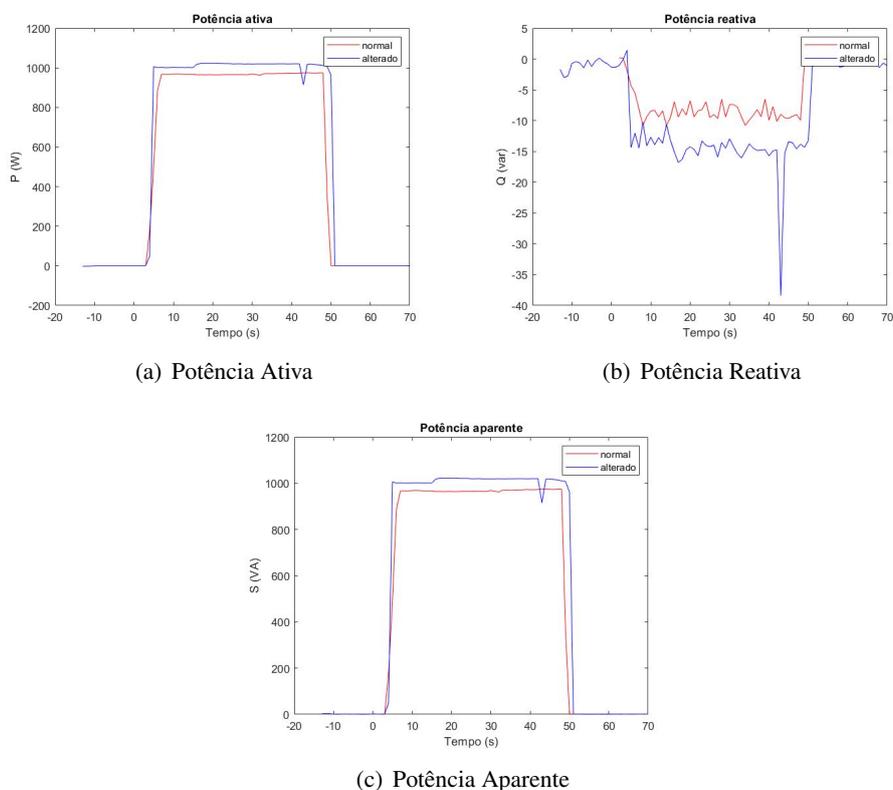


Figura 5.6: Gráficos comparativo dos parâmetros elétricos do ensaio com obstrução total da saída de ar

5.4.4 Ensaio secador de cabelo em tensão 110V

Um ensaio foi realizado utilizando tensão de rede de 110V com o devido ajuste no secador de cabelo. Neste ensaio o resistor de entrada do circuito de alimentação do módulo queimou e não foi possível obter resultados. Com este resultado considera-se que este módulo não poderia ser utilizados em localidades onde este nível de tensão 110V predomina. Seria necessário realizar adaptações no circuito de alimentação

do SA9903b de modo a adaptar o módulo para o nível de tensão.

Capítulo 6

Conclusões e Perspectivas

A questão que norteou este trabalho era de qual maneira um sistema de medição de consumo de energia elétrica, em tempo real, poderia contribuir para a redução de perdas. A aplicação do módulo desenvolvido em um trabalho de conclusão anterior (NEPOMUCENO; COZENDEY, 2013) era o objetivo deste trabalho.

A realização de medições visa utilizar os recursos disponíveis da forma eficiente, procurando detectar e prever falhas no sistema. Estes testes foram realizados para testar a resposta do sistema a diferentes entradas em uma aplicação real.

O sistema supervisorio desenvolvido no ScadaBr que compõe os três módulos mostrou que funcionam separadamente satisfazendo os propósitos iniciais. Os módulos são responsáveis por monitorar os parâmetros relacionados a energia elétrica, o responsável pelas medições e o módulo responsável pelo recebimento de dados, apresentaram resultados aplicáveis em sistemas de automação predial.

Na última etapa, com a exibição dos testes e seus resultados, podemos confirmar ainda mais o sucesso parcial do objetivo. Uma forma de identificar falhas em tempo real por transmissão sem fio foi utilizado, mas não foi implementado um sistema que permite informar ao usuário que ocorreram falhas.

Sendo assim, é fácil notar que o sistema, com todas as suas funcionalidades, poderá contribuir para uma economia de energia. É interessante lembrar que os componentes empregados no desenvolvimento dos módulos e os *softwares* utilizados são *open source* e materiais de baixo custo tornando o sistema viável economicamente para aplicações reais.

6.1 Trabalhos Futuros

Dentro da automação predial, segue-se pela busca por soluções de eficiência energética. Como trabalhos futuros podem ser exploradas as seguintes áreas:

- Utilização de CI's alternativos, além do Sames, mais atualizado;
- Busca de uma alternativa para substituição do Resistor *Shunt*, de modo a melhorar a segurança contra contato com o usuário e choques. Além disso, buscar melhores formas de isolar o módulo com a criação de uma estrutura para envolver o módulo;

- Implementação dos módulos com *ZigBit* disponíveis no LARA para implementação de uma rede com o padrão *ZigBee* ;
- Ou desenvolvimento de uma rede *Internet of Things* (IoT) com módulos de medidores e atuadores;
- Testes de consumo de aparelhos de ar condicionado ao longo do tempo levando em consideração mais parâmetros, tais como temperatura externa, umidade, número de indivíduos no ambiente;
- Aplicação dos módulos para controle e identificação de falhas em aparelhos de ar condicionado;
- Estudo sobre desenvolvimento de uma rede neural visando o aprendizado do sistema com relação ao comportamento dos equipamentos;
- Implementação e realização de testes de atuação com relé de estado sólido;
- Busca por uma solução para utilização do módulo medidor de energia em tensões de 110V e 220V;

REFERÊNCIAS BIBLIOGRÁFICAS

- ACC, P. E. d. M. *O que é calibração e sua importancia no processo e na qualidade!* São José dos Pinhais, PR: [s.n.], 2015. Disponível em: <<http://www.accpr.com.br/o-que-e-calibracao-e-sua-importancia-no-processo-e-na-qualidade/>>. Acesso em: Junho 2018.
- BOYER, S. A. *SCADA supervisory control and data acquisition*. Carolina do Norte, Estados Unidos: The Instrumentation, Systems and Automation Society, 2018.
- BRONZATTI, F. L.; IAROSZINSKI, A. Matrizes energéticas no brasil: cenário 2010-2030. *Encontro Nacional de Engenharia de Produção*, v. 28, p. 13–16, 2008.
- CARDOZO, A.; OLIVEIRA, I. N.; STRUMINSKI, E. Aplicabilidade de sistemas operacionais de tempo real (rtos) para automação residencial. *Caderno de Graduação-Ciências Exatas e Tecnológicas-UNIT-ALAGOAS*, v. 4, n. 1, 2017.
- COLEMAN, H. W.; STEELE, W. G. *Experimentation, validation, and uncertainty analysis for engineers*. Nova Jersey, Estados Unidos: John Wiley & Sons, 2018.
- CREDER, H. Instalações elétricas. 15°. *Rio de Janeiro: LTC*, 2010.
- EUBAC. *European building automation controls association*. Bruxelas: online, 2017. Disponível em: <<http://www.eubac.org/certification-labelling-audits/system-audits/index.html>>. Acesso em: Dezembro 2017.
- IRWIN, D. J.; NELMS, R. M. *Análise Básica de Circuitos Para Engenharia*. Rio de Janeiro: Grupo Gen-LTC, 2000.
- KOPKE, J.; ZLOCCOWICK, J. H.; SCHREIER, S. Conservação de energia e combate ao desperdício. *ANEEL*, 2015.
- LINKERSPITE. *Thermal Module*. Disponível em: <http://linksprite.com/wiki/index.php5?title=Thermal_Module>.
- MARQUES, M. C. S.; HADDAD, J.; GUARDIA, E. C. a. Eficiência energética: teoria & prática. *Itajubá, FUPAI*, 2007.
- MODBUS. Modbus organization, inc. *online*, online, 2014. Disponível em: <<http://www.modbus.org/faq.php>. [Dostopano12.12.2016]>.
- NAM, U. V. Transforming our world: The 2030 agenda for sustainable development. 2015.
- NEPOMUCENO, A. L. S.; COZENDEY, G. C. *Desenvolvimento de um Módulo de Medição de Energia Wireless com Transmissor em Tempo Real de Parâmetros para Cargas de até 9kW*. Monografia (Trabalho de Graduação em Engenharia de Controle e Automação) — Universidade de Brasília, Brasília, DF, 2013.

NUNES, D. R. *Ferramentas e instrumentos de medidas elétricas*. Campos dos Goytacazes (RJ): Essentia Editora, 2010.

OLIVEIRA, R.; FARINES, J.; FRAGA, J. *Sistemas de tempo-real*. São Paulo: IME-USP, 2000.

POLITERM. *Tecnologia em Instrumentos de Medição*. online, 2016. Disponível em: <<http://www.politerm.com.br/Produto-PRODUTOS-Wattímetros>>. Acesso em: Outubro 2017.

PROCELINFO. Selo procel edificações de eficiência energética. online, 2015. Disponível em: <<http://www.procelinfo.com.br/>>. Acesso em: Setembro 2017.

SABERELETRICA. *Saiba Verificar o bom funcionamento do medidor de energia elétrica e de sua instalação*. online, 2016. Disponível em: <<https://www.sabereletrica.com.br/medidor-de-energia-eletrica/>>. Acesso em: Outubro 2017.

SAMES. *SA9903B datasheet-Single Phase Power/Energy IC with SPI Interface*. Africa do Sul.

SCADABR. Sistema open-source para supervisão e controle. online, 2014. Disponível em: <<http://www.scadabr.com.br/>>. Acesso em: Maio 2018.

TMP35, L. V. T. S. *Tmp36/tmp37 datasheet*, © 1996–2010 analog devices. Inc. *All rights reserved*.

WANG, S. *Intelligent buildings and building automation*. Estados Unidos e Canadá: Routledge, 2009.

ANEXOS

I. DIAGRAMAS ESQUEMÁTICOS

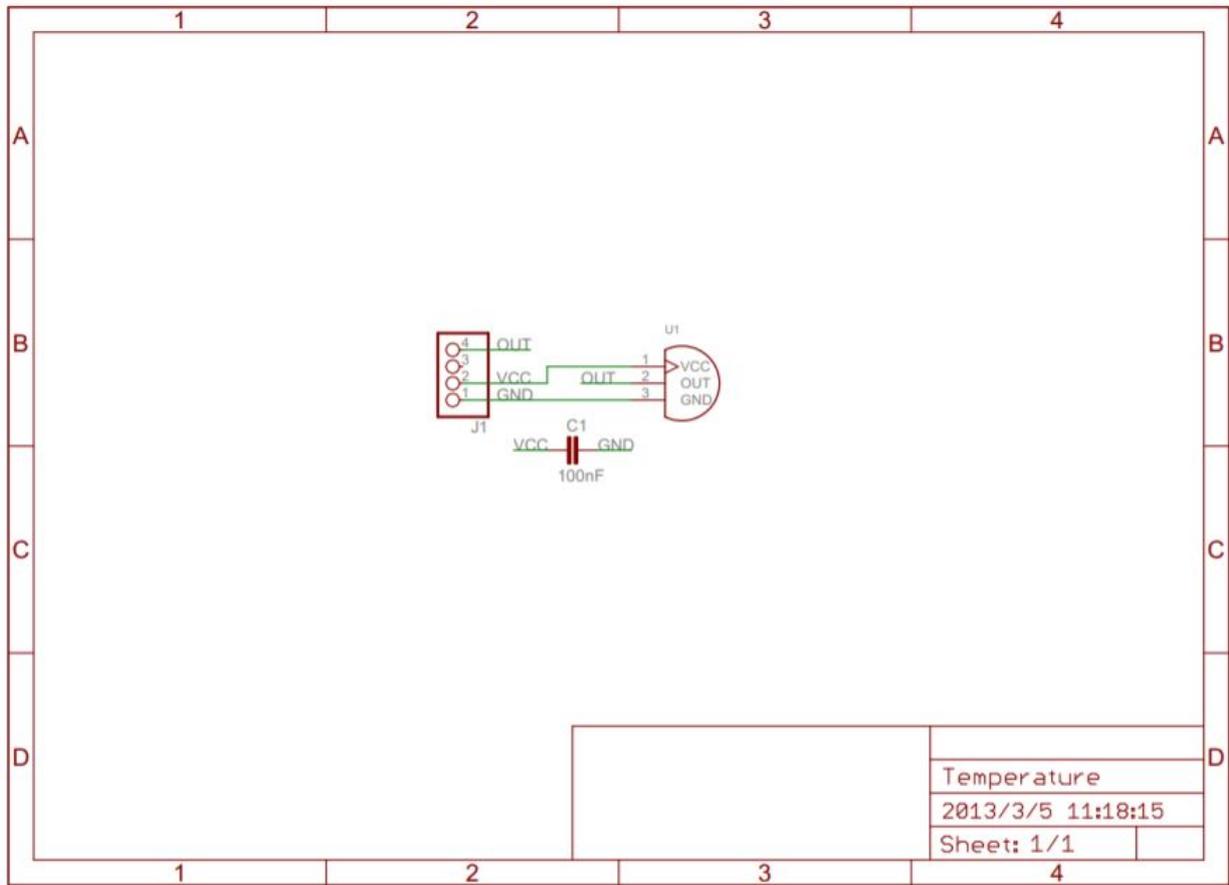


Figura I.1: Esquemático Módulo de temperatura
Fonte:(LINKERSPITE,)

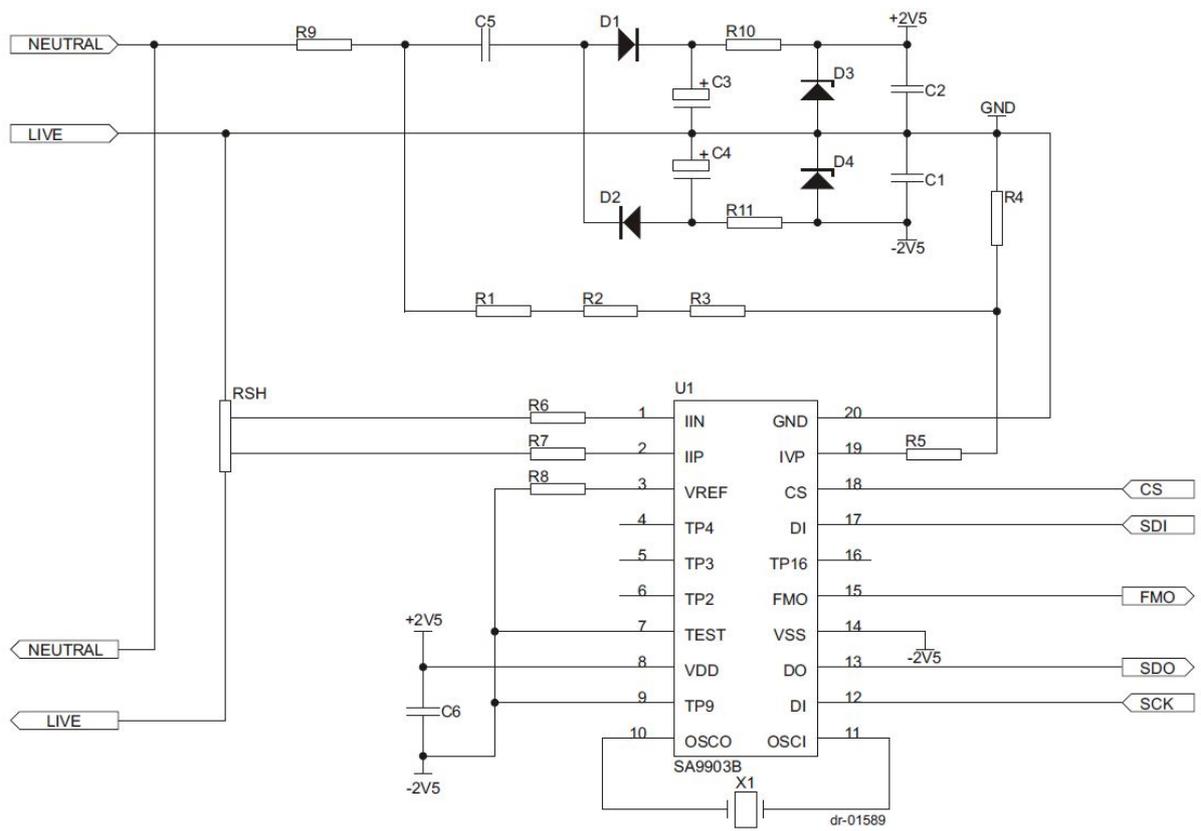


Figura I.2: Esquemático de aplicação sugerida do SA9903B da SAMES
 Fonte:(SAMÉS,)

II. TABELAS DOS COMPONENTES

Tabela II.1: Tabela para o circuito de aplicação sugerida do SA9903B da SAMES

Symbol	Description
U1	SA9903B
R1	Resistor, 110k, 1/4W, 1% metal
R2	Resistor, 110k, 1/4W, 1% metal
R3	Resistor, 130k, 1/4W, 1% metal
R4	Resistor, 24k, 1/4W, 1% metal
R5	Resistor, 1M, 1/4W, 1% metal
R6	Resistor, 1.6k, 1/4W, 1% metal
R7	Resistor, 1.6k, 1/4W, 1%, metal
R8	Resistor, 24K, 1/4W, 1%, metal
R9	Resistor, 47R, 2W, 5%, wire wound
R10	Resistor, 680R, 1/4W, 1%, metal
R11	Resistor, 680R, 1/4W, 1%, metal
RSH	Shunt Resistor, 80A/50mV
C1	Capacitor, 220nF, ceramic
C2	Capacitor, 220nF, ceramic
C3	Capacitor, 220uF, 16V, electrolytic
C4	Capacitor, 220uF, 16V, electrolytic
C5	Capacitor, 470nF, 250VAC, polyester
C6	Capacitor, 820nF, ceramic
D1	Diode, 1N4003
D2	Diode, 1N4003
D3	Diode, Zener, 2.5V
D4	Diode, Zener, 2.5V
X1	Crystal, 3.579545MHz

Fonte:(SAMES,)

III. CÓDIGOS

III.1 Código Utilizado para medidor de temperatura

```
1
2 //Codigo para implementar Sensor de Temperatura
3 //com integracao com Modbus para o ScadaBR
4 //Fonte: linksprite.com/wiki/index.php5?title=Thermal_Module & embarcados.com.br/
   arduino-e-scadabr/
5
6
7 #include <SimpleModbus.h>
8
9 //temperature sensor and modbus protocol
10 //TMP36 Pin Variables
11
12
13
14 int sensorPin = A0;
15 //the analog pin the TMP36s Vout (sense) pin is connected to
16 //the resolution is 10 mV / degree centigrade with a
17         //500 mV offset to allow for negative temperatures
18
19 /*
20 * setup() - this function runs once when you turn your Arduino on
21 * We initialize the serial connection with the computer
22 */
23 enum
24 {
25
26     TEMPERATURE,
27     HOLDING_REGS_SIZE
28 };
29
30 unsigned int holdingRegs[HOLDING_REGS_SIZE];
31
32
33 void setup()
34 {
35     // Serial.begin(9600); //Start the serial connection with the computer
36         //to view the result open the serial monitor
37
38
39     modbus_configure(&Serial, 9600, SERIAL_8N1, 1, 2, HOLDING_REGS_SIZE, holdingRegs);
40
41     modbus_update_comms(9600, SERIAL_8N1, 1);
42 }
43
```

```

44 void loop()                                // run over and over again
45 {
46     modbus_update();
47
48     //getting the voltage reading from the temperature sensor
49     int reading = analogRead(sensorPin);
50
51     // converting that reading to voltage , for 3.3v arduino use 3.3
52     float voltage = reading * 5.0;
53     voltage /= 1024.0;
54
55     // print out the voltage
56     //Serial.print(voltage); Serial.println(" volts");
57
58     // now print out the temperature
59     float temperatureC = (voltage - 0.5) * 100 ; //converting from 10 mv per degree wit
60         500 mV offset                                //to degrees ((voltage - 500mV) times
61                                                         100)
62
63
64
65     Serial.print(temperatureC); Serial.println(" degrees C");
66
67     // now convert to Fahrenheit
68     //float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;
69     // Serial.print(temperatureF); Serial.println(" degrees F");
70
71
72     delay(1000);                                //waiting a second
73
74     //Begin of Modbus section
75
76
77
78     holdingRegs[TEMPERATURE] = analogRead(temperatureC);
79
80
81 }

```

Listing III.1: ModTemp.cpp

III.2 Código Utilizado para medidor de Energia Elétrica

```

1
2 /*
3 Autores: Andre Luiz Siega Nepomuceno e Gabriel Calache Cozendey
4 Email: andrerm2k@hotmail.com / gcozendey@outlook.com
5

```

```

6 Este codigo esta livre para copia.
7 */
8
9 #include "sa9903b.h"
10 #include <SPI.h> //nao sei pq tem que dar include aqui tb, ja que ja existe um
    include em sa9903b.cpp
11 // #include <SimpleModbusSlave.h>
12 #include <SimpleModbus.h>
13 //Numero de identificacao do no escravo.
14 #define SLAVE_ID 1
15
16 //Descricao das variaveis no registrador holding. Note que REGISTER_SIZE apenas
    define o tamanho desse vetor.
17 enum {ERRORS,VOLTAGE,FREQUENCY,ACTIVE,REACTIVE,REGISTER_SIZE,HOLDING_REGS_SIZE};
18
19 SA9903B sa9903b;
20
21 unsigned int reg[REGISTER_SIZE] = {0};
22 int volt;
23
24 unsigned int holdingRegs[HOLDING_REGS_SIZE];
25 void setup ()
26 {
27     // Serial.begin(57600);
28     sa9903b.Setup();
29     modbus_configure(&Serial, 57600, SERIAL_8N1, 1, 2, HOLDING_REGS_SIZE, holdingRegs);
30 }
31 void loop ()
32 {
33     sa9903b.Run();
34
35
36     // Atualiza os registradores de holding e MODBUS
37
38     if (sa9903b.GetUpdatedStatus())
39     {
40         //Voltage
41         reg[VOLTAGE] = sa9903b.GetVoltage();
42         // Serial.print( reg[VOLTAGE]); Serial.println(" volts");
43         // delay(1000);
44
45         //Frequency
46         reg[FREQUENCY] = sa9903b.GetFrequency();
47
48         // Serial.println( reg[FREQUENCY]); Serial.println(" hertz");
49         delay(1000);
50
51         //Active Power
52         reg[ACTIVE] = abs(sa9903b.GetActive());
53         // Serial.print( reg[ACTIVE]); Serial.println(" W");
54         // delay(1000);
55

```

```

56 //Reactive Power
57 reg[REACTIVE] = abs(sa9903b.GetReactive());
58 // Serial.print(reg[REACTIVE]); Serial.println(" var");
59 // delay(1000);
60 //
61 //Error
62 reg[ERRORS] = modbus_update(reg);
63 // Serial.print(reg[ERRORS]); Serial.println(" er");
64 // delay(1000);
65 }
66
67 }

```

Listing III.2: aquisicaosa9903b

```

1 #include "sa9903b.h"
2 #include <SPI.h>
3
4 //Inicializa variaveis
5 SA9903B::SA9903B()
6 {
7     active = 0;
8     last_active = 0;
9     reactive = 0;
10    last_reactive = 0;
11    frequency = 0;
12    voltage = 0;
13
14    last_fmo = LOW;
15
16    last_time = 0;
17    elapsed_time = 0;
18
19    updated = false;
20 }
21
22 SA9903B::~~SA9903B()
23 {
24 }
25 }
26
27 //Inicializa comunicacao SPI e pinos de passagem por zero e chip select
28 void SA9903B::Setup()
29 {
30     SPI.begin();
31     SPI.setBitOrder(MSBFIRST);
32     SPI.setClockDivider(SPI_CLOCK_DIV64);
33
34     pinMode(FMO_PIN, INPUT);
35
36     pinMode(CS_PIN, OUTPUT);
37     digitalWrite(CS_PIN, HIGH);

```

```

38 }
39
40 /*
41 Le um registrador no endereco address.
42
43 O registrador do SAMES e composto por 3 bytes. A leitura e feita byte a byte e o
    resultado e armazenado na variavel result atravez de sucessivos deslocamento e
    copia de bits.
44 A logica da comunicacao e INVERSA pois os bits que atravessam o optoacoplador sao
    invertidos.
45 A cada leitura o pino de CHIP SELECT deve ser ativado e desativado no final da
    operacao.
46 */
47 long SA9903B::Read(const byte address)
48 {
49     long result = 0;
50
51     digitalWrite(CS_PIN,LOW);
52
53     SPI.setDataMode(SPI_MODE2);
54     SPI.transfer(~START_ADDR);
55     SPI.transfer(~address);
56
57     SPI.setDataMode(SPI_MODE3);
58     result = SPI.transfer(0xFF);
59     result = result << 8;
60     result |= SPI.transfer(0xFF);
61     result = result << 8;
62     result |= SPI.transfer(0xFF);
63
64     digitalWrite(CS_PIN,HIGH);
65
66     return ~result;
67 }
68
69 //Le todos os registradores de forma sequencial
70 void SA9903B::ReadAll()
71 {
72     active = Read(ACTIVE_ADDR);
73     reactive = Read(REACTIVE_ADDR);
74     frequency = Read(FREQUENCY_ADDR);
75     voltage = Read(VOLTAGE_ADDR);
76 }
77
78 //Traduz e retorna o valor da tensao
79 float SA9903B::GetVoltage()
80 {
81     voltage &= 0x00FFFFFF;
82     return RATED_VOLTAGE/700.0*voltage;
83
84
85 }

```

```

86
87 //Traduz e retorna o valor da frequencia
88 float SA9903B::GetFrequency()
89 {
90     frequency &= 0x000003FF;
91     return CRYSTAL_FREQUENCY/256.0/frequency;
92 }
93
94 /*
95 Traduz e retorna o valor da potencia ativa.
96
97 Nesa funcao e necessario que se armazene o intervalo de tempo entre leituras
98 consecutivas pois o medidos de potencia e na verdade um integrador.
99 Alem disso, como o registrador de potencia possui limitacoes, o mesmo pode sofrer um
100 overflow e retornar para zero (ou para seu valor maximo, no caso inverso).
101 Tais condicoes devem ser verificadas e corrigidas.
102 */
103 float SA9903B::GetActive()
104 {
105     active &= 0x00FFFFFF;
106     float time = elapsed_time/1000000.0;
107     float diff;
108
109     if (active < 0x100000 && last_active > 0xEFFFFFFF)
110     {
111         diff = (active+0xFFFFFFFF)-last_active;
112     }
113     else if (active > 0xEFFFFFFF && last_active < 0x100000)
114     {
115         diff = active -(last_active+0xFFFFFFFF);
116     }
117     else
118     {
119         diff = active -last_active;
120     }
121
122     return RATED_VOLTAGE*RATED_CURRENT*diff/time/320000.0;
123 }
124
125 /*
126 Traduz e retorna o valor da potencia reativa.
127
128 Nesa funcao e necessario que se armazene o intervalo de tempo entre leituras
129 consecutivas pois o medidos de potencia e na verdade um integrador.
130 Alem disso, como o registrador de potencia possui limitacoes, o mesmo pode sofrer um
131 overflow e retornar para zero (ou para seu valor maximo, no caso inverso).
132 Tais condicoes devem ser verificadas e corrigidas.
133 */
134 float SA9903B::GetReactive()
135 {
136     reactive &= 0x00FFFFFF;
137     float time = elapsed_time/1000000.0;

```

```

134 float diff;
135
136 if (reactive < 0x100000 && last_reactive > 0xEFFFFFFF)
137 {
138     diff = (reactive+0xFFFFFFFF)-last_reactive;
139 }
140 else if (reactive > 0xEFFFFFFF && last_reactive < 0x100000)
141 {
142     diff = reactive -(last_reactive+0xFFFFFFFF);
143 }
144 else
145 {
146     diff = reactive-last_reactive;
147 }
148
149 return RATED_VOLTAGE*RATED_CURRENT*diff/time/320000.0;
150 }
151
152 //Funcao principal que deve ser executada a cada ciclo do programa principal. Detecta
153 //uma passagem por zero e realiza a leitura dos registradores do SA9903B
154 void SA9903B::Run()
155 {
156     int fmo = digitalRead(FMO_PIN);
157
158     if (fmo == HIGH && last_fmo == LOW) //verifica se ocorreu passagem por zero da
159     //tensao
160     {
161         last_active = active;
162         last_reactive = reactive;
163
164         unsigned long time = micros();
165
166         ReadAll();
167
168         elapsed_time = time - last_time; // calcula tempo entre as leituras
169         last_time = time;
170
171         updated = true;
172     }
173
174     last_fmo = fmo;
175 }
176
177 //Retorna TRUE caso as leituras tenham sido atualizadas
178 boolean SA9903B::GetUpdatedStatus()
179 {
180     if (updated)
181     {
182         updated = false;
183         return true;
184     }
185     return false;

```

Listing III.3: adquisicaosa9903b.cpp

```
1
2 #ifndef SA9903B_H_GUARD
3 #define SA9903B_H_GUARD
4
5 #include <Arduino.h>
6
7 #define START_ADDR 0x01
8 #define ACTIVE_ADDR 0x80
9 #define REACTIVE_ADDR 0x81
10 #define VOLTAGE_ADDR 0x82
11 #define FREQUENCY_ADDR 0x83
12
13 #define CS_PIN 48
14 #define FMO_PIN 49
15
16 // #define RATED_VOLTAGE 230.0
17 // #define RATED_CURRENT 40.0
18 // #define RATED_VOLTAGE 235.013
19 #define CORRECTION_FACTOR 230.0/265.0
20 #define RATED_VOLTAGE 230.0*CORRECTION_FACTOR
21 #define RATED_CURRENT 39.68
22 #define CRYSTAL_FREQUENCY 3579545.0
23
24 class SA9903B
25 {
26 public:
27     SA9903B();
28     ~SA9903B();
29
30     void Setup();
31     void Run();
32
33     float GetVoltage();
34     float GetFrequency();
35     float GetActive();
36     float GetReactive();
37     boolean GetUpdatedStatus();
38     long Read(byte address);
39 private:
40     long active;
41     long last_active;
42     long reactive;
43     long last_reactive;
44     long frequency;
45     long voltage;
46
47     int last_fmo;
48
```

```

49 unsigned long last_time;
50 unsigned long elapsed_time;
51
52 boolean updated;
53
54
55 void ReadAll();
56 };
57
58 #endif //SA9903B_H_GUARD

```

Listing III.4: aquisicaosa9903b.h

```

1
2
3 #include "SimpleModbusSlave.h"
4
5 #define BUFFER_SIZE 128
6
7 // frame [] is used to receive and transmit packages.
8 // The maximum serial ring buffer size is 128
9 unsigned char frame[BUFFER_SIZE];
10 unsigned int holdingRegsSize; // size of the register array
11 unsigned char broadcastFlag;
12 unsigned char slaveID;
13 unsigned char function;
14 unsigned char TxEnablePin;
15 unsigned int errorCount;
16 unsigned int T1_5; // inter character time out
17 unsigned int T3_5; // frame delay
18
19 // function definitions
20 void exceptionResponse(unsigned char exception);
21 unsigned int calculateCRC(unsigned char bufferSize);
22 void sendPacket(unsigned char bufferSize);
23
24 unsigned int modbus_update(unsigned int *holdingRegs)
25 {
26     unsigned char buffer = 0;
27     unsigned char overflow = 0;
28
29     while (Serial.available())
30     {
31         // The maximum number of bytes is limited to the serial buffer size of 128 bytes
32         // If more bytes is received than the BUFFER_SIZE the overflow flag will be set
33         // and the
34         // serial buffer will be red untill all the data is cleared from the receive
35         // buffer.
36         if (overflow)
37             Serial.read();
38         else
39             {

```

```

38     if (buffer == BUFFER_SIZE)
39         overflow = 1;
40     frame[buffer] = Serial.read();
41     buffer++;
42 }
43 delayMicroseconds(T1_5); // inter character time out
44 }
45
46 // If an overflow occurred increment the errorCount
47 // variable and return to the main sketch without
48 // responding to the request i.e. force a timeout
49 if (overflow)
50     return errorCount++;
51
52 // The minimum request packet is 8 bytes for function 3 & 16
53 if (buffer > 7)
54 {
55     unsigned char id = frame[0];
56
57     broadcastFlag = 0;
58
59     if (id == 0)
60         broadcastFlag = 1;
61
62     if (id == slaveID || broadcastFlag) // if the recieved ID matches the slaveID or
63         // broadcasting id (0), continue
64     {
65         unsigned int crc = ((frame[buffer - 2] << 8) | frame[buffer - 1]); // combine
66         // the crc Low & High bytes
67         if (calculateCRC(buffer - 2) == crc) // if the calculated crc matches the
68         // recieved crc continue
69         {
70             function = frame[1];
71             unsigned int startingAddress = ((frame[2] << 8) | frame[3]); // combine the
72             // starting address bytes
73             unsigned int no_of_registers = ((frame[4] << 8) | frame[5]); // combine the
74             // number of register bytes
75             unsigned int maxData = startingAddress + no_of_registers;
76             unsigned char index;
77             unsigned char address;
78             unsigned int crc16;
79
80             // broadcasting is not supported for function 3
81             if (!broadcastFlag && (function == 3))
82             {
83                 if (startingAddress < holdingRegsSize) // check exception 2 ILLEGAL DATA
84                 // ADDRESS
85                 {
86                     if (maxData <= holdingRegsSize) // check exception 3 ILLEGAL DATA VALUE
87                     {
88                         unsigned char noOfBytes = no_of_registers * 2;
89                         unsigned char responseFrameSize = 5 + noOfBytes; // ID, function ,

```

```

84         noOfBytes, (dataLo + dataHi) * number of registers, crcLo, crcHi
85     frame[0] = slaveID;
86     frame[1] = function;
87     frame[2] = noOfBytes;
88     address = 3; // PDU starts at the 4th byte
89     unsigned int temp;
90
91     for (index = startingAddress; index < maxData; index++)
92     {
93         temp = holdingRegs[index];
94         frame[address] = temp >> 8; // split the register into 2 bytes
95         address++;
96         frame[address] = temp & 0xFF;
97         address++;
98     }
99
100    crc16 = calculateCRC(responseFrameSize - 2);
101    frame[responseFrameSize - 2] = crc16 >> 8; // split crc into 2 bytes
102    frame[responseFrameSize - 1] = crc16 & 0xFF;
103    sendPacket(responseFrameSize);
104 }
105 else
106     exceptionResponse(3); // exception 3 ILLEGAL DATA VALUE
107 }
108 else
109     exceptionResponse(2); // exception 2 ILLEGAL DATA ADDRESS
110 }
111 else if (function == 16)
112 {
113     // check if the recieved number of bytes matches the calculated bytes minus
114     // the request bytes
115     // id + function + (2 * address bytes) + (2 * no of register bytes) + byte
116     // count + (2 * CRC bytes) = 9 bytes
117     if (frame[6] == (buffer - 9))
118     {
119         if (startingAddress < holdingRegsSize) // check exception 2 ILLEGAL DATA
120         ADDRESS
121         {
122             if (maxData <= holdingRegsSize) // check exception 3 ILLEGAL DATA VALUE
123             {
124                 address = 7; // start at the 8th byte in the frame
125
126                 for (index = startingAddress; index < maxData; index++)
127                 {
128                     holdingRegs[index] = ((frame[address] << 8) | frame[address + 1]);
129                     address += 2;
130                 }
131
132                 // only the first 6 bytes are used for CRC calculation
133                 crc16 = calculateCRC(6);
134                 frame[6] = crc16 >> 8; // split crc into 2 bytes
135                 frame[7] = crc16 & 0xFF;

```

```

132
133         // a function 16 response is an echo of the first 6 bytes from the
           request + 2 crc bytes
134         if (!broadcastFlag) // dont respond if its a broadcast message
           sendPacket(8);
135     }
136     else
137         exceptionResponse(3); // exception 3 ILLEGAL DATA VALUE
138     }
139     else
140         exceptionResponse(2); // exception 2 ILLEGAL DATA ADDRESS
141     }
142     else
143         errorCount++; // corrupted packet
144     }
145     else
146         exceptionResponse(1); // exception 1 ILLEGAL FUNCTION
147     }
148     else // checksum failed
149         errorCount++;
150     } // incorrect id
151 }
152 }
153 else if (buffer > 0 && buffer < 8)
154     errorCount++; // corrupted packet
155
156 return errorCount;
157 }
158
159 void exceptionResponse(unsigned char exception)
160 {
161     errorCount++; // each call to exceptionResponse() will increment the errorCount
162     if (!broadcastFlag) // dont respond if its a broadcast message
163     {
164         frame[0] = slaveID;
165         frame[1] = (function | 0x80); // set the MSB bit high, informs the master of an
           exception
166         frame[2] = exception;
167         unsigned int crc16 = calculateCRC(3); // ID, function + 0x80, exception code == 3
           bytes
168         frame[3] = crc16 >> 8;
169         frame[4] = crc16 & 0xFF;
170         sendPacket(5); // exception response is always 5 bytes ID, function + 0x80,
           exception code, 2 bytes crc
171     }
172 }
173
174 void modbus_configure(long baud, unsigned char _slaveID, unsigned char _TxEnablePin,
           unsigned int _holdingRegsSize)
175 {
176     slaveID = _slaveID;
177     Serial.begin(baud);
178

```

```

179 if (_TxEnablePin > 1)
180 { // pin 0 & pin 1 are reserved for RX/TX. To disable set txenpin < 2
181   TxEnablePin = _TxEnablePin;
182   pinMode(TxEnablePin, OUTPUT);
183   digitalWrite(TxEnablePin, LOW);
184 }
185
186 // Modbus states that a baud rate higher than 19200 must use a fixed 750 us
187 // for inter character time out and 1.75 ms for a frame delay.
188 // For baud rates below 19200 the timing is more critical and has to be calculated
189
190 // E.g. 9600 baud in a 10 bit packet is 960 characters per second
191 // In milliseconds this will be 960characters per 1000ms. So for 1 character
192 // 1000ms/960characters is 1.04167ms per character and finally modbus states an
193 // intercharacter must be 1.5T or 1.5 times longer than a normal character and thus
194 // 1.5T = 1.04167ms * 1.5 = 1.5625ms. A frame delay is 3.5T.
195
196 if (baud > 19200)
197 {
198   T1_5 = 750;
199   T3_5 = 1750;
200 }
201 else
202 {
203   T1_5 = 15000000/baud; // 1T * 1.5 = T1.5
204   T3_5 = 35000000/baud; // 1T * 3.5 = T3.5
205 }
206
207 holdingRegsSize = _holdingRegsSize;
208 errorCount = 0; // initialize errorCount
209 }
210
211 unsigned int calculateCRC(byte bufferSize)
212 {
213   unsigned int temp, temp2, flag;
214   temp = 0xFFFF;
215   for (unsigned char i = 0; i < bufferSize; i++)
216   {
217     temp = temp ^ frame[i];
218     for (unsigned char j = 1; j <= 8; j++)
219     {
220       flag = temp & 0x0001;
221       temp >>= 1;
222       if (flag)
223         temp ^= 0xA001;
224     }
225   }
226   // Reverse byte order.
227   temp2 = temp >> 8;
228   temp = (temp << 8) | temp2;
229   temp &= 0xFFFF;
230   return temp; // the returned value is already swopped - crcLo byte is first & crcHi

```

```

    byte is last
230 }
231
232 void sendPacket(unsigned char bufferSize)
233 {
234     if (TxEnablePin > 1)
235         digitalWrite(TxEnablePin, HIGH);
236
237     for (unsigned char i = 0; i < bufferSize; i++)
238         Serial.write(frame[i]);
239
240     Serial.flush();
241
242     // allow a frame delay to indicate end of transmission
243     delayMicroseconds(T3_5);
244
245     if (TxEnablePin > 1)
246         digitalWrite(TxEnablePin, LOW);
247 }

```

Listing III.5: SimpleModbusSlave.cpp

```

1 #ifndef SIMPLE_MODBUS_SLAVE_H
2 #define SIMPLE_MODBUS_SLAVE_H
3
4 /*
5 SimpleModbusSlave allows you to communicate
6 to any slave using the Modbus RTU protocol.
7
8 The crc calculation is based on the work published
9 by jpmzometa at
10 http://sites.google.com/site/jpmzometa/arduino-mbrt
11
12 By Juan Bester : bester.juan@gmail.com
13
14 The functions implemented are functions 3 and 16.
15 read holding registers and preset multiple registers
16 of the Modbus RTU Protocol, to be used over the Arduino serial connection.
17
18 This implementation DOES NOT fully comply with the Modbus specifications.
19
20 Specifically the frame time out have not been implemented according
21 to Modbus standards. The code does however combine the check for
22 inter character time out and frame time out by incorporating a maximum
23 time out allowable when reading from the message stream.
24
25 These library of functions are designed to enable a program send and
26 receive data from a device that communicates using the Modbus protocol.
27
28 SimpleModbusSlave implements an unsigned int return value on a call to modbus_update
29 ().
30
31 This value is the total error count since the slave started. Its useful for fault

```

finding.

This code is for a Modbus slave implementing functions 3 and 16
function 3: Reads the binary contents of holding registers (4X references)
function 16: Presets values into a sequence of holding registers (4X references)

All the functions share the same register array.

Exception responses:

- 1 ILLEGAL FUNCTION
- 2 ILLEGAL DATA ADDRESS
- 3 ILLEGAL DATA VALUE

Note:

The Arduino serial ring buffer is 128 bytes or 64 registers.
Most of the time you will connect the arduino to a master via serial
using a MAX485 or similar.

In a function 3 request the master will attempt to read from your
slave and since 5 bytes is already used for ID, FUNCTION, NO OF BYTES
and two BYTES CRC the master can only request 122 bytes or 61 registers.

In a function 16 request the master will attempt to write to your
slave and since a 9 bytes is already used for ID, FUNCTION, ADDRESS,
NO OF REGISTERS, NO OF BYTES and two BYTES CRC the master can only write
118 bytes or 59 registers.

Using the FTDI converter ic the maximum bytes you can send is limited
to its internal buffer which is 60 bytes or 30 unsigned int registers.

Thus:

In a function 3 request the master will attempt to read from your
slave and since 5 bytes is already used for ID, FUNCTION, NO OF BYTES
and two BYTES CRC the master can only request 54 bytes or 27 registers.

In a function 16 request the master will attempt to write to your
slave and since a 9 bytes is already used for ID, FUNCTION, ADDRESS,
NO OF REGISTERS, NO OF BYTES and two BYTES CRC the master can only write
50 bytes or 25 registers.

Since it is assumed that you will mostly use the Arduino to connect to a
master without using a USB to Serial converter the internal buffer is set
the same as the Arduino Serial ring buffer which is 128 bytes.

The functions included here have been derived from the
Modbus Specifications and Implementation Guides

http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf

http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf

http://www.modbus.org/docs/PI_MBUS_300.pdf

*/

```
81
82 #include "Arduino.h"
83
84 // function definitions
85 void modbus_configure(long baud, byte _slaveID, byte _TxEnablePin, unsigned int _
      holdingRegsSize);
86 unsigned int modbus_update(unsigned int *holdingRegs);
87
88
89 #endif
```

Listing III.6: SimpleModbusSlave.h