

WELMER SALES DE SOUSA

**DESENVOLVIMENTO DE UM SISTEMA WEB DE MONITORAMENTO
DE CONSUMO DE ENERGIA ELÉTRICA**

Manaus

2021

WELMER SALES DE SOUSA

**DESENVOLVIMENTO DE UM SISTEMA WEB DE MONITORAMENTO
DE CONSUMO DE ENERGIA ELÉTRICA**

Projeto de Pesquisa desenvolvido durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de bacharel em Engenharia de Controle e Automação.

Orientador: Moisés Pereira Bastos, Me.

Manaus

2021

CIP – Catalogação na Publicação

SOUSA, Welmer Sales de

Desenvolvimento De Um Sistema Web De Monitoramento De Consumo De Energia Elétrica / Welmer Sales de Sousa. – Manaus, 2021.

69p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia de Controle e Automação). Universidade do Estado do Amazonas, 2021.

Orientador: Moisés Pereira Bastos.

1. Sistema Web. 2. Monitoramento. 3. Remoto. I. BASTOS, Moisés Pereira. II. Desenvolvimento De Um Sistema Web De Monitoramento De Consumo De Energia Elétrica.

WELMER SALES DE SOUSA

DESENVOLVIMENTO DE UM SISTEMA WEB DE MONITORAMENTO DE CONSUMO DE ENERGIA ELÉTRICA

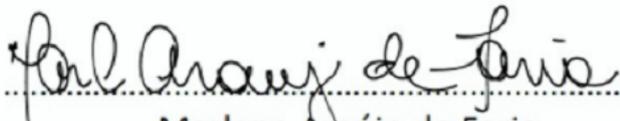
Pesquisa desenvolvida durante a disciplina Trabalho de Conclusão de Curso II apresentada à banca avaliadora do Curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Estado do Amazonas como pré-requisito para a obtenção do título de Bacharel em Engenharia de Controle e Automação.

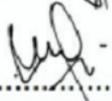
Nota obtida: 8,66 (Oito, oitenta e seis)

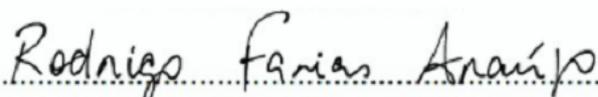
Aprovada em 20/12/2021.

Área de concentração: Engenharia Eletrônica

BANCA EXAMINADORA


.....
Marlene Araújo de Faria


.....
Israel Mazaira Morales


.....
Rodrigo Farias Araújo

Manaus

2021

*Aos meus familiares, amigos e todos
aqueles que me acompanharam nos
momentos mais adversos e que mesmo
longe estão sempre comigo.*

AGRADECIMENTOS

Agradeço imensamente aos meus pais por darem a oportunidade e recursos para que eu concluísse meu curso. Agradeço aos meus amigos por estarem sempre ao meu lado me apoiando e sendo os melhores amigos que eu poderia ter nesse momento. Agradeço a todos os professores do curso de Engenharia de Controle e Automação por quem tenho grande admiração e com quem aprendi muito nessa longa caminhada na graduação. E por fim, agradeço profundamente à Universidade do Estado do Amazonas por toda a estrutura de pessoal e de instalações oferecida por meio de investimentos públicos que propiciou a minha formação nesses anos de curso.

*“Se a aparência e a essência das coisas
coincidissem, a ciência seria
desnecessária.”*

(Karl Marx)

RESUMO

Hoje, a eletricidade é algo essencial no cotidiano das pessoas, mas a geração de energia elétrica exige a exploração de muitos recursos naturais e tem alto custo de produção, que acaba sendo repassado aos cidadãos. Assim, o presente trabalho teve como principal objetivo o desenvolvimento de um sistema de monitoramento de consumo de energia elétrica residencial não invasivo que permitisse ao usuário estimar seus gastos com energia elétrica. Para a elaboração do projeto proposto, utilizou-se Arduino e sensor de corrente não invasivo SCT013-000 para a verificação do consumo de energia elétrica, para o compartilhamento das informações utilizou-se o Ethernet *Shield* e para a leitura foi desenvolvida uma página HTML que pode ser acessada através de um computador ou dispositivo móvel. Ao final do projeto obteve-se uma plataforma web que proporcionou ao usuário a possibilidade de monitorar de maneira remota os seus gastos com energia elétrica trabalho.

Palavras-chave: monitoramento; plataforma web; remota.

ABSTRACT

Today, electricity is essential in people's daily lives, but the generation of electricity requires the exploitation of many natural resources and has a high production cost, which ends up being passed on to citizens. Thus, the present work had as main objective the development of a non-invasive residential electric energy consumption monitoring system that allows the user to estimate his electric energy expenses. For the elaboration of the proposed project, an Arduino and a non-invasive current sensor SCT013-000 were used to verify the consumption of electric energy, the Ethernet Shield was used for the sharing of information, and for reading an HTML page was developed that can be accessed through a computer or mobile device. At the end of the project, a web platform was obtained that provided the user with the possibility of remotely monitoring their electricity costs.

Key-words: monitoring; web platform; remotely.

LISTA DE FIGURAS

Figura 1 - Medidor de energia elétrica eletromecânico.....	20
Figura 2 - Arduino UNO.....	23
Figura 3 - Sensor de corrente SCT013-000.....	24
Figura 4 - Ethernet <i>Shield</i> W5100.....	25
Figura 5 - Fluxo de funcionamento do projeto.....	28
Figura 6 - Diagrama de funcionamento do projeto.....	29
Figura 7 - Diagrama de Funcionamento do Sistema.....	32
Figura 8 - Sensor de Corrente com <i>plug</i> P2.....	35
Figura 9 - Esquema do resistor de carga.....	36
Figura 10 - Divisor de tensão.....	36
Figura 11 - Circuito <i>offset</i>	37
Figura 12 - Bibliotecas utilizadas.....	38
Figura 13 - Parâmetros de comunicação.....	39
Figura 14 - Comando para inicialização do servidor.....	39
Figura 15 - Configuração do sensor no Arduino.....	39
Figura 16 - Calibração da entrada do Arduino.....	40
Figura 17 - Cálculo do consumo de energia.....	40
Figura 18 - Texto HTML da página.....	41
Figura 19 - Página projetada para o Sistema.....	41
Figura 20 - Sensor de corrente com sua saída adaptada.....	42
Figura 21 - Circuito de medição montado.....	43
Figura 22 - Resultado da medição de corrente no monitor serial (lâmpada 70W).....	44
Figura 23 - Medição de corrente do Amperímetro (lâmpada 70W).....	44
Figura 24 - Resultado da medição de corrente no monitor serial (Ventilador).....	45
Figura 25 - Medição de corrente do Amperímetro (Ventilador).....	45
Figura 26 - Resultado da medição de corrente no monitor serial (Ferro de passar).....	46

Figura 27 - Medição de corrente do Amperímetro (Ferro de Passar).....	46
Figura 28 - Mensagem de teste do Ethernet <i>Shield</i>	47
Figura 29 - Mensagem de alerta do Arduino.....	48
Figura 30 - Página de supervisão projetada.....	48
Figura 31 - Página do sistema.....	49
Figura 32 - Circuito completo do sistema.....	49
Figura 33 - Página indicando que o sistema está ligado.....	50
Figura 34 - Página mostrando resultado de medição (Sanduicheira primeiro teste).....	50
Figura 35 - Medição da corrente da sanduicheira pelo amperímetro.....	51
Figura 36 - Página mostrando resultado de medição (Sanduicheira – segundo teste).....	51
Figura 37 - Resultado de medição (Ferro de passar – primeiro teste).....	52
Figura 38 - Medição da corrente do ferro de passar roupa pelo amperímetro (primeiro teste).....	52
Figura 39 - Resultado de medição (Ferro de passar – segundo teste).....	53
Figura 40 - Medição da corrente do ferro de passar roupa pelo amperímetro (segundo teste).....	53
Figura 41 - Resultado de medição (Ventilador – primeiro teste).....	54
Figura 42 - Medição da corrente do ventilador pelo amperímetro (segundo teste).....	54
Figura 43 - Resultado de medição (Ventilador – segundo teste).....	55
Figura 44 - Medição da corrente do ventilador pelo amperímetro (segundo teste).....	55
Figura 45 – Interface projetada para o sistema.....	58

LISTA DE QUADROS

Quadro 1 - Comparativo entre os valores obtidos pelo sensor e pelo amperímetro.....	47
Quadro 2 - Comparativo entre os resultados do sistema e o estimado por cálculo.....	56
Quadro 3 - Custo do projeto.....	56

SUMÁRIO

1 INTRODUÇÃO	14
1.1 TEMA.....	15
1.2 PROBLEMAS DE PESQUISA.....	15
1.3 HIPÓTESE.....	15
1.4 OBJETIVOS.....	15
1.4.1 Objetivo geral	15
1.4.2 Objetivos específicos	16
1.5 JUSTIFICATIVA.....	16
1.6 METODOLOGIA.....	17
1.7 ORGANIZAÇÃO DO TRABALHO.....	18
2 REFERENCIAL TEÓRICO	19
2.1 MEDIDORES DE ENERGIA ELÉTRICA.....	19
2.2 SISTEMAS SUPERVISÓRIOS.....	20
2.3 MICROCONTROLADORES.....	21
2.3.1 Arduino	22
2.4 SENSORES DE CORRENTE NÃO INVASIVO SCT013.....	23
2.5 ETHERNET <i>SHIELD</i>	25
2.6 HTML.....	26
2.7 SERVIDORES WEB.....	27
2.8 TRABALHOS RELACIONADOS.....	28
2.8.1 Sistema Para Monitoramento De Consumo De Energia Elétrica Particular, Em Tempo Real E Não Invasivo Utilizando A Tecnologia Arduino	28
2.8.2 medidor de demanda de energia elétrica residencial com acesso remoto	29
2.8.3 Sistema Supervisório Para Acompanhamento De Consumo De Energia Elétrica Residencial	29
3 MATERIAIS E MÉTODOS	31
3.1 DISPOSITIVO PROPOSTO.....	31
3.1.1 Implementação do Sistema	31
3.1.2 Hardware	33

3.1.2.1 Sensor SCT013-000.....	33
3.1.2.2 O circuito de medição e ajuste do sinal.....	35
3.1.2.3 Arduino Uno, Ethernet <i>Shield</i> e a comunicação.....	37
3.1.3 Software.....	38
3.1.3.1 Bibliotecas.....	38
3.1.3.2 Código de comunicação.....	38
3.1.3.3 Calibração e leitura do sensor.....	39
3.1.3.4 Programa para cálculo do consumo.....	40
3.1.3.5 Página web.....	41
4 RESULTADOS.....	42
4.1 CIRCUITO DE MEDIÇÃO.....	42
4.2 COMUNICAÇÃO DO SISTEMA.....	47
4.3 O SISTEMA EM FUNCIONAMENTO.....	49
4.4 CUSTOS PARA O DESENVOLVIMENTO DO PROJETO.....	56
5 CONCLUSÃO.....	57
5.1 DIFICULDADES ENCONTRADAS.....	58
5.2 TRABALHOS FUTUROS.....	59
REFERÊNCIAS BIBLIOGRÁFICAS.....	60

1 INTRODUÇÃO

Hoje, a eletricidade é algo essencial no cotidiano das pessoas, principalmente nos grandes centros urbanos. A sociedade contemporânea vive conectada e a energia elétrica é fundamental para manter as pessoas ligadas ao mundo, mas, para que a eletricidade chegue até a população, muitos recursos naturais são explorados e a produção de energia elétrica acaba sendo danosa ao meio ambiente e com custos elevados, e quem acaba pagando essa conta são os cidadãos.

O crescimento do desperdício de energia elétrica resulta obrigatoriamente no aumento da potência instalada de geração. Este acréscimo de potência se reflete numa elevação de custos, tanto ambientais quanto em investimentos em equipamentos (PAULA, 2013).

De acordo com informações do PROCEL (2019) aproximadamente 29% da energia produzida no Brasil é consumida em residências, e, embora o PROCEL relate reduções no desperdício de energia elétrica nos últimos anos, esses números podem ser melhorados. Ter um consumo mais consciente, monitorando os gastos, e mudar alguns hábitos cotidianos como tomar banhos curtos, apagar luzes ao sair dos aposentos, não deixar eletrodomésticos ligados, nem abrir a porta da geladeira sem necessidade podem contribuir para uma maior redução do desperdício.

Observando este cenário, tem sido cada vez mais frequentes discursos e debates evidenciando a questão da sustentabilidade na sociedade, não só buscando formas alternativas e menos agressivas de produção de energia elétrica, mas também em busca de formas de conscientizar a população em prol de um consumo mais responsável para que se diminua o desperdício. Nesse sentido, um consumo mais consciente de energia elétrica é fundamental dentro do contexto da sustentabilidade, pois grande parte da energia consumida mundo atualmente vem de fontes não renováveis.

O sistema de monitoramento de consumo energia elétrica proposto nesse projeto de pesquisa trabalha em cima dessa questão da conscientização, sendo uma ferramenta que pode auxiliar o combate ao desperdício. O dispositivo a ser desenvolvido propõe-se não somente à realização da medição do consumo, mas também se dispõe a possibilitar visualização dos dados de consumo remotamente através de um aplicativo para dispositivos móveis, oferecendo um maior conforto visto que os medidores, ou contadores, de energia elétrica residencial convencionais

apresentam apenas uma leitura local. Assim, o projeto busca facilitar o controle do consumo de energia elétrica residencial e, conseqüentemente, contribuir para a redução do desperdício e de gastos desnecessários.

1.1 TEMA

Desenvolvimento de um sistema web de monitoramento de consumo de energia elétrica.

1.2 PROBLEMA DE PESQUISA

A necessidade de disponibilizar às pessoas um sistema de medição de consumo de energia elétrica que facilite (ou simplifique) a obtenção de informações sobre os gastos mensais com eletricidade e também ofereça um maior conforto e segurança por meio do acesso remoto com notificações sobre o status do fornecimento de energia na residência em tempo real.

1.3 HIPÓTESE

É possível criar um sistema de monitoramento de consumo de energia elétrica, utilizando o *kit* de prototipagem Arduino, sensor de corrente não invasivo e o Ethernet *Shield*, que por meio de uma página HTML informe aos usuários as taxas de consumo após cada medição realizada, possibilitando que o usuário estipule seus gastos e tenha um consumo mais sustentável.

1.4 OBJETIVOS

1.4.1 Objetivo geral

Desenvolver um sistema supervisor que obtenha informações da rede elétrica residencial através de um sensor de corrente não invasivo e processe essas informações utilizando o kit de prototipagem Arduino, facilitando o monitoramento do consumo de energia elétrica local. Os dados processados pelo sistema serão enviados a um servidor usando internet por meio do Ethernet *Shield* e serão

disponibilizados ao usuário em uma página HTML, podendo o acesso às informações do sistema ser feito no computador ou em um dispositivo móvel.

1.4.2 Objetivos específicos

- Realizar levantamento bibliográfico;
- Estudar as tecnologias a serem utilizadas no desenvolvimento do projeto;
- Desenvolver circuito para medição do consumo de energia elétrica;
- Realizar compartilhamento dos dados utilizando internet;
- Desenvolver da página HTML para supervisão do sistema;
- Testar funcionalidades do dispositivo desenvolvido.

1.5 JUSTIFICATIVA

O que impulsionou a realização desta pesquisa foi a possibilidade de oferecer à sociedade um maior conforto e segurança no que diz respeito ao consumo de energia elétrica, visando conceder às pessoas não só informações sobre seus gastos, mas também notificar os usuários sobre o status do fornecimento de energia elétrica na residência, proporcionando uma maior autonomia e conhecimento sobre o consumo de energia elétrica. Assim, a pesquisa busca dar uma alternativa às pessoas para que estas possam ter um maior controle sobre o consumo de energia elétrica em suas residências e que através da utilização do projeto proposto possam ter um consumo mais consciente, reduzindo o desperdício e contribuindo, assim, para um mundo mais sustentável.

A realização desta pesquisa possibilita a aplicação e apresentação dos conhecimentos obtidos durante os anos de graduação no curso de Engenharia de Controle e Automação, tendo como foco principalmente os seguintes componentes curriculares: Sistemas Microprocessados, Eletrônica Analógica e Digital, Redes de Computadores, Automação e Supervisão de Processos e Linguagem de Programação.

1.6 METODOLOGIA

O presente trabalho representa uma Pesquisa Aplicada que tem como objetivo a realização de uma pesquisa exploratória sobre o conteúdo bibliográfico, bem como de laboratório, adquiridos sobre o assunto. Os procedimentos técnicos utilizados são os de pesquisa bibliográfica e experimental. O método de abordagem utilizado é o hipotético-dedutivo e a elaboração segue o método de procedimento monográfico. Para coleta de dados são utilizados documentação indireta, com auxílio de documentos primários e secundários, e a análise e interpretação de seus dados será qualitativa.

Primeiramente, foram realizadas pesquisas bibliográficas na área de sistemas microprocessados, circuitos eletrônicos e programação para aplicações de dispositivos móveis. A partir dessa pesquisa, foram estudadas as tecnologias utilizadas no desenvolvimento do projeto.

Definidos os componentes a foram utilizados no desenvolvimento do projeto, deu-se início à construção do protótipo do sistema, começando com a elaboração do circuito que realiza a medição do consumo de energia elétrica. Para o desenvolvimento deste circuito foram utilizados o sensor de corrente não invasivo SCT013-000, que obtém as informações do consumo de energia, os dados obtidos são processados pelo microcontrolador ATMEGA328, um dos microcontroladores que constituem o Arduino Uno.

Concluída a elaboração do circuito de medição, desenvolveu-se o circuito que possibilita o compartilhamento de dados usando a internet. Para tal foi utilizado o Ethernet *Shield*. Outra etapa do projeto foi o desenvolvimento da página HTML por meio da qual é possível ter acesso às informações obtidas pelo sistema através de um computador ou dispositivo móvel. Para auxiliar a criação desta página foi utilizado o software Notepad++, que é uma ferramenta que facilita o desenvolvimento de códigos nas mais diversas linguagens de programação. E então, realizou-se a integração das três partes do projeto.

Por fim, testes foram realizados para efetuação de análises quanto ao desempenho do protótipo, a fim de sempre buscar uma otimização do projeto desenvolvido, corrigindo eventuais erros de funcionamento observados.

1.7 ORGANIZAÇÃO DO TRABALHO

No capítulo 1 deste trabalho são descritos os seguintes itens: introdução, problema de pesquisa, hipótese, objetivo, justificativa e a metodologia que foi utilizada para conclusão desse trabalho. O Capítulo 2 mostra o referencial teórico da monografia, com as pesquisas relacionadas com o tema do trabalho, para o melhor entendimento do leitor, bem como os trabalhos relacionados ao tema abordado nesta monografia, expondo suas principais características para realização de um comparativo entre os mesmos e o trabalho apresentado. O Capítulo 3 tem como título “Materiais e métodos”, nesse capítulo, são descritos os principais materiais utilizados no decorrer do trabalho, assim como os métodos e técnicas utilizados para sua conclusão. O Capítulo 4 tem como título “Resultados” onde são expostos os principais resultados obtidos na conclusão deste trabalho. No Capítulo 5 está a conclusão da monografia que apresenta as dificuldades encontradas e sugere trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 MEDIDORES DE ENERGIA ELÉTRICA

Os contadores de energia elétrica convencionais são dispositivos eletromecânicos que são utilizados pela concessionária, empresa responsável pelo fornecimento de energia elétrica, para aferir o consumo de energia elétrica nas residências. Este instrumento de medição foi inventado por Oliver Blackburn Shallenberger, em 1888, baseado num medidor de ampère-hora para corrente alternada, passando posteriormente a ser adotado como padrão pela indústria (PAULA, 2013).

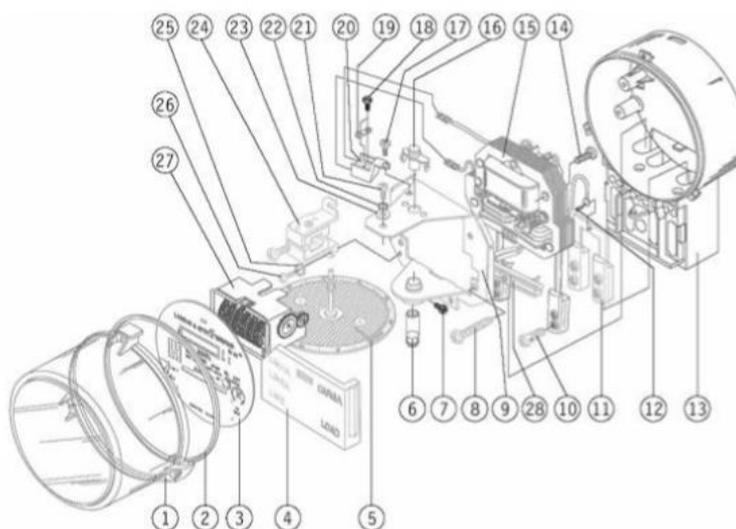
Hoje, praticamente todas as residências possuem um medidor de energia elétrica, uma vez que o consumo de energia elétrica precisa ser mensurado com a maior exatidão possível para que a cobrança por parte das concessionárias de energias seja justa com os usuários. É estimado atualmente que 92% dos medidores são eletromecânicos, esse modelo é mais comumente utilizado por possui baixo custo e alta disponibilidade. O erro máximo desses aparelhos é de 2% de seu valor nominal de operação e essa taxa de erro é considerada relativamente alta levando-se em conta todas as perdas que podem gerar prejuízos para o fornecedor e/ou para o consumidor (BRITO, 2016).

O medidor de energia eletromecânico encontrado na maioria das residências é constituído fundamentalmente por duas partes: uma é a eletromecânica e a outra é um sistema de relojoaria. Inicialmente, nos medidores de energia elétrica em que ocorria disposição de metal em um eletrodo, bastava multiplicar a corrente elétrica por um fator envolvendo a tensão de linha para encontrar o valor da energia. Esse fator adquiriu uma nova forma de cálculo a partir de 1903 que é mantida até hoje (PAULA, 2013).

Mesmo com os aprimoramentos ocorridos ao longo do tempo, a indução se manteve como mais importante princípio de funcionamento. A maior parte dos consumidores brasileiros seja residencial, comercial ou industrial ainda tem a medição de seu consumo realizado pelos medidores eletromecânicos do tipo indução (PAULA, 2013).

Um diagrama esquemático é apresentado na Figura 1, mostrando as partes que constituem um medidor eletromecânico monofásico e a descrição de cada item é indicado.

Figura 1 – Medidor de energia elétrica eletromecânico.



Item	Descrição	Item	Descrição
1	Tampa do medidor	15	Elemento motor
2	Gaxeta da tampa do Medidor	16	Mancal superior (pino guia)
3	Placa de identificação	17	Parafuso fixação superior do terminal de prova
4	Tampa do bloco terminais	18	Parafuso fixação do gancho de prova
5	Elemento Móvel	19	Gancho de prova
6	Mancal inferior (magnético)	20	Suporte prova interna
7	Parafuso fixação do mancal inferior	21	Parafuso fixação do registrador
8	Parafuso fixação do elemento motor	22	Arruela pressão
9	Armadura	23	Arruela lisa
10	Parafuso terminal de corrente	24	Imã frenador
11	Terminal ligação de neutro	25	Arruela pressão
12	Parafuso fixação do terminal de potencial-ligação de neutro	26	Parafuso fixação do conjunto imã
13	Base e bloco do medidor	27	Registrador
14	Parafuso fixação do elemento armadura	28	Suporte da indutiva

Fonte: (PAULA, 2013, p. 17)

2.2 SISTEMAS SUPERVISÓRIOS

Sistema Supervisório é um sistema digital de monitoramento utilizado para o gerenciamento de variáveis relacionadas a um determinado processo. Todas as informações obtidas e coletadas são constantemente atualizadas com o funcionamento do processo e podem ser armazenadas em um banco de dados local, ou remoto, para fins de registro histórico e/ou posterior uso, como, por exemplo, para elaboração de relatórios, planilhas, gráficos, variando de acordo com

o interesse do usuário. Os sistemas supervisórios também são conhecidos também por *Supervisory Control and Data Acquisition* (SCADA) (SOUZA; NUNES; BIANCHINI, 2016).

Os sistemas SCADA são aplicações que possibilitam que valores de variáveis de um processo produtivo ou instalação física sejam coletadas, interpretadas, registradas e oferecidas ao usuário, podendo essas informações ser monitoradas por meio de telas gráficas com atualizações instantâneas das variáveis do processo. Um sistema supervisório é composto basicamente por esses elementos: sensores e atuadores, rede de comunicação, estações remotas (aquisição/controle) e de monitoração central (AZEVEDO, 2013).

O sistema SCADA possui cinco elementos em sua arquitetura:

- Operador: Operador Humano que monitora o sistema SCADA e executa funções de controle de supervisão para as operações da planta remotos.
- Interface homem-máquina (IHM): apresenta dados para o operador e prevê entradas de controle numa variedade de formatos, incluindo gráficos, esquemas, janelas, menus suspensos, telas sensíveis ao toque, etc.
- Unidade Terminal Master (MTU): Equivalente a uma unidade mestre em uma arquitetura mestre/escravo. A MTU apresenta dados para o operador através da IHM, reúne dados a partir do local distante, e transmite sinais de controle para o local remoto.
- Meios de comunicação: método de comunicação entre a MTU e controladores remotos. A comunicação pode ser através da Internet, redes com ou sem fios, ou a rede telefônica pública comutada.
- Unidade Terminal Remota (RTU): Funciona como um escravo na arquitetura mestre/escravo. Envia sinais de controle para o dispositivo controlado, adquire dados a partir desses dispositivos, e transmite os dados para o MTU. Um RTU pode ser um CLP (Krutz, 2006, p.7).

Sistemas supervisórios podem ser usados para o monitoramento de uma variedade de informações dos mais distintos e complexos processos e é interessante perspectiva de utilização de um sistema supervisório para monitoramento do consumo de energia elétrica como uma alternativa para que se contenha o desperdício de energia elétrica.

2.3 MICROCONTROLADORES

Antes de entender o que é um microcontrolador, é necessário primeiramente saber o que é um microprocessador, para que não haja confusão entre as

definições. Microprocessador é um componente eletrônico, mas propriamente um circuito integrado, desenvolvido para realização de tarefas específicas com uma linguagem de comando específica. É, falando de forma resumida, a parte de circuito responsável pela realização de cálculos e tomada de decisão (NICOLOSI, 2013).

Falando dos microcontroladores, suas principais características estão relacionadas ao fato desses dispositivos reunirem em um único *chip* todos os periféricos que são necessários para o projeto e fabricação de aparelhos eletrônicos dos mais diversos tipos, desde simples sinalizadores e luzes pisca-pisca até equipamentos médicos sofisticados. De forma mais objetiva, enquanto os microprocessadores são circuitos integrados responsáveis unicamente pelo processamento de dados, o microcontrolador é um dispositivos que reúne no seu chip um microcontrolador mais memórias, *timers* e outros periféricos, sendo mais completos (PEREIRA, 2013).

2.3.1 Arduino

O Arduino é uma placa, ou *kit* de prototipagem, de desenvolvimento de projetos criada em 2005 pelo italiano Massimo Banzi com o intuito de contribuir no aprendizado de eletrônica para estudantes. O Arduino é uma plataforma de código aberto (*open source*) que tem como um de seus principais objetivos oferecer uma plataforma de baixo custo para que estudantes pudessem desenvolver seus projetos e protótipos com o menor custo possível (BRITO, 2016).

Pode-se utilizar o Arduino tanto para desenvolvimento de objetos interativos independentes, quanto para ser conectado a um computador, a uma rede, ou até mesmo à Internet para envio e recebimento de dados, e atuação sobre os mesmos. Assim, o Arduino pode ser utilizado para compartilhar dados recebidos de sensores, por exemplo, enviando-os para um site, através do qual esses dados poderão ser exibidos na forma de gráfico (MCROBERTS, 2011).

O Arduino UNO, apresentado na Figura 2, é a placa mais popular entre os projetistas, este modelo possui um microcontrolador ATMEGA328 e um ATMEGA16U2, além de outros componentes eletrônicos. O primeiro microcontrolador é ligado a todas as portas de entrada e saída e alimentação disponíveis na placa, além das portas *In-System Programming* (ICSP), utilizadas para a instalação do *bootloader* (é um programa gravado na memória do micro

controlador que simplifica o carregamento de programas para o *chip* de memória flash) no microcontrolador. Já o segundo microcontrolador tem todas as ligações necessárias para uso da interface USB, além da ICSP de mesma função da primeira (BARRETO, 2015).

A tensão de operação do Arduino Uno é de 5 volts, a plataforma dispõe de um cristal de 16MHz para geração da frequência (*clock*) em que as instruções são executadas. A placa também possui uma memória *Flash* de 32 Kb (0,5 Kb reservados ao *bootloader*) para armazenamento de códigos de programas, 2 Kb de SRAM e 1 Kb para a EEPROM. 20 pinos de E/S, sendo 14 deles digitais (6 com PWM) e 6 analógicos estão disponíveis para utilização. O Arduino UNO possui ainda um *buffer* de dados na serial de 64 bytes, com uma fila utilizando o escalonamento *round robin* (BARRETO, 2015).

Figura 2 – Arduino UNO.



Fonte: (THOMSEN, 2014)

Sobre a programação, a IDE do Arduino possui uma linguagem própria baseada na linguagem C e C++ (MCROBERTS, 2011).

2.4 SENSOR DE CORRENTE NÃO INVASIVO SCT013-000

O SCT013 é um sensor de corrente não invasivo, que permite a medição de corrente elétrica sem que seja necessário inseri-lo no circuito elétrico, com um funcionamento similar a um alicate amperímetro. Esse sensor é composto fundamentalmente por um transformador de corrente e um resistor de referência, é

possível encontrar vários modelos desse sensor variam de acordo com o valor de corrente nominal máximo de entrada que o sensor consegue ler, podendo ser de 5A, 10A, 15 A, 20 A, 25 A, 30 A, 50 A, 60 A e 100 A, e com saída em tensão de 0 a 1 V. O modelo de 100 que possui saída em corrente (GOMES; GOMES; TAGLIARI, 2016).

O Sensor SCT013-000, apresentado na Figura 3, é o modelo dessa família de sensores que, segundo sua especificação técnica, suporta na entrada uma corrente de até 100A, com saída de 50mA no secundário, possuindo um fator K de 2000 (que está relacionada às espiras do trafo) e tolerando temperaturas que podem alternar entre -25°C até 70°C . A faixa de operação do sensor é não linear em 3% e possui uma taxa de variação de seu valor nominal que vai se 10% a 120%, equivalente a 10A até 120A. Isso quer dizer que essa é a melhor faixa para se trabalhar (BRITO, 2016).

Figura 3 – Sensor de corrente SCT013-000.



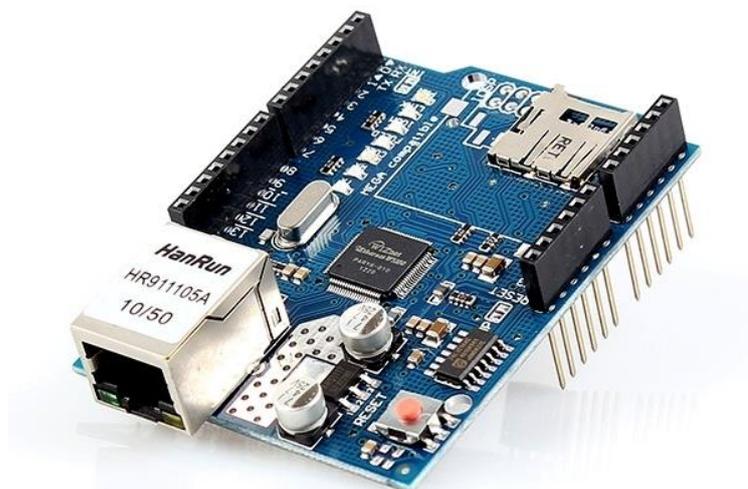
Fonte: (THOMSEN, 2015)

Pelo fato do SCT013-000 possuir na saída uma variação de corrente, diferentemente dos outros modelos menos robustos que possuem uma variação de tensão que é necessária para que o Arduino entenda os dados lidos, e gerar uma variação de tensão a partir da saída para que a leitura possa ser feita pelo Arduino. Essa variação pode ser feita através de um resistor de carga, também denominado como “Burden Resistor” (BRITO, 2016).

2.5 ETHERNET SHIELD

O Ethernet *Shield*, mostrado na Figura 4, permite que uma placa Arduino se conecte a internet. Ele é baseado no chip de ethernet Wiznet W5100 provendo um *stack* de rede (ponte entre Arduino e rede) que suporta os protocolos TCP e UDP. Além de conectar o Arduino à rede, o Ethernet *Shield* possui um *slot* na placa de cartão micro-SD, o qual pode ser usado para armazenar arquivos como banco de dados para a rede e é compatível com o Arduino Uno e Mega. O Arduino se comunica tanto com o *shield* quanto com o cartão SD usando o barramento SPI (*Serial Peripheral Interface* é um protocolo que possibilita a comunicação do microcontrolador com outros componentes). Tanto o Ethernet *Shield* como a função SDCard compartilham o barramento SPI, então somente um pode estar ativo de cada vez. (BRITO,2016)

Figura 4 - Ethernet Shield W5100



Fonte: (THOMSEN, 2014)

O funcionamento do *Shield* se baseia na W5100 Wiznetchip ethernet. O Wiznet W5100 fornece uma pilha com suporte aos protocolos de rede TCP e UDP (IP) Ele suporta até quatro conexões simultâneas. O Ethernet *Shield* se conecta a

uma placa Arduino usando conectores barra de pinos longos que se estendem através do *Shield* permitindo que outros *Shields* possam ser empilhados em cima (PAULA, 2013).

O *Shield* também inclui um controlador de reset, para garantir que o módulo Ethernet W5100 está devidamente reiniciado quando necessário, já que revisões anteriores do *Shield* não eram compatíveis com o mega e precisava ser reiniciado manualmente após a inicialização. Uma série de LEDs informativos são encontrados no *shield* para acompanhamento do seu funcionamento: PWR (indica que a placa e o *shield* estão alimentados), LINK (indica a presença de um link de rede e pisca quando o *shield* transmite ou recebe dados), FULLD (indica que a conexão de rede é *full duplex*), 100M (indica a presença de uma ligação de rede 100 Mb), RX (sinaliza quando o *shield* recebe dados) TX (sinaliza quando o *shield* envia dados), COLL (indica quando são detectadas colisões de rede). (PAULA, 2013)

2.6 HTML

HTML é a sigla para *HyperText Markup Language*, que, em português, significa linguagem para marcação de hipertexto. Hipertexto é todo texto inserido em um documento para a web e que tem como principal característica a possibilidade de se interligar a outros documentos da web com uso dos nossos já conhecidos links, presentes nas páginas dos sites que estamos acostumados a visitar. Então, todo o conteúdo textual que você vê em uma página de um site é um hipertexto, assim como imagens, vídeos, gráficos, sons e conteúdos não textuais em geral são chamados de hipermídia. Quando a HTML foi inventada, os conteúdos eram essencialmente hipertextos, com a hipermídia surgindo posteriormente. Assim, hoje, a HTML é uma linguagem para marcação de conteúdos web em geral. (SILVA, 2015)

Desde sua criação, os idealizadores da HTML tiveram a preocupação de retirar da linguagem de marcação qualquer atribuição ou função de apresentação, ou seja, HTML destina-se exclusivamente a estruturar documentos. É nessa destinação que se fundamentam os princípios básicos do desenvolvimento seguindo os Padrões Web. (SILVA, 2015)

Documento HTML, documento web, página web, ou simplesmente documento, é como são conhecidos os arquivos completo resultado de um código

desenvolvido em HTML, contendo toda a marcação necessária para ser lida e interpretada por um dispositivo de usuário. As ferramentas utilizadas para o desenvolvimento de hipertextos são conhecidas como editores HTML. Basicamente, pode-se escrever marcação HTML usando qualquer editor de texto, como o Word, disponível em ambiente Windows, por exemplo, porém é mais adequada a utilização de editores de texto próprios para escrever marcação HTML. (SILVA, 2015).

2.7 SERVIDOR WEB

Os servidores web são fundamentais atualmente, já que por meio da utilização desses servidores é possível realizar o armazenamento e a troca de informações com outra máquina. Os servidores web são responsáveis por oferecer um serviço baseado na arquitetura cliente-servidor, onde os computadores clientes fazem solicitações ao computador servidor, que então organiza essas requisições solicitadas, processa de acordo com suas métricas e responde para o computador que solicitou a devida tarefa. O cliente, geralmente, usa os navegadores como “Mozilla Firefox” e o “Google Chrome” para fazer a solicitação e o servidor trabalha com softwares para realizar a requisição solicitada. Para essa troca de informação utiliza-se o protocolo *Hypertext Transfer Protocol* (HTTP) de comunicação da Web (SOUZA, 2013).

Servidores Web, normalmente, são responsáveis por gerar páginas web no formato HTML para exibição nos navegadores de internet dos usuários. Porém, os servidores podem ser usados também para exibirem respostas em outros formatos, como por exemplo, XML e JSON. Esses outros formatos provêm uma padronização simples que facilita na comunicação entre o servidor e aplicações mais leves, como requisições solicitadas por aplicativos móveis (SOARES, 2016).

É muito comum na elaboração de aplicativos a conexão com um servidor web para realização de consultas e armazenamento de dados, colocando a cargo do servidor as requisições de processamento mais pesado, deixando, dessa forma, os aplicativos mais leves (SOARES, 2016).

Há uma variedade de servidores disponíveis para uso, entre eles os mais conhecidos são: IIS (*Internet Information Services*) da Microsoft, o Xitami, que é servidor *open source* desenvolvido como HTTP pessoal, e o Apache HTTP Server, o servidor mais robusto no mercado. (SOUZA, 2013)

2.8 TRABALHOS RELACIONADO

Neste capítulo são apresentados e explanados trabalhos que tem relação com o tema proposto nesta monografia. Foi realizado um estudo visando apresentar suas principais características.

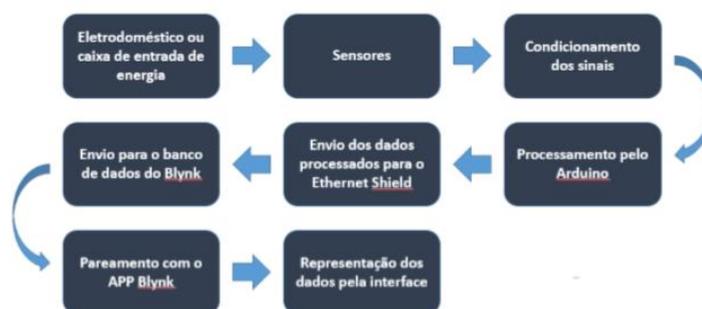
2.8.1 Sistema Para Monitoramento De Consumo De Energia Elétrica Particular, Em Tempo Real E Não Invasivo Utilizando A Tecnologia Arduino

Na monografia de Brito (2016), o autor aborda de maneira clara a necessidade de termos uma maior conscientização no consumo de energia elétrica, procurando facilitar o acesso das pessoas às informações sobre o seu consumo de energia elétrica e conseqüentemente buscando diminuir o desperdício de energia elétrica.

O objetivo de Brito (2016) foi desenvolver um dispositivo eletrônico que realiza a leitura da corrente elétrica utilizando um sensor não invasivo e partir da leitura da corrente elétrica calcula o consumo de energia elétrica, visando oferecer às pessoas uma ferramenta de baixo custo que auxilie os consumidores a controlar os gastos.

Brito (2016) utiliza o *kit* de prototipagem Arduino UNO para o processamento dos dados obtidos pelo sensor não invasivo SCT 013-000 e usa o *Shield* Ethernet para compartilhamento dos dados via internet. Para a leitura dos dados o autor optou pela utilização do aplicativo Blink, uma plataforma desenvolvida para iOS e Android para controlar Arduino e outros dispositivos. A figura 5 mostra o fluxo de funcionamento do projeto.

Figura 5 – Fluxo de funcionamento do projeto



Fonte: (BRITO, 2016)

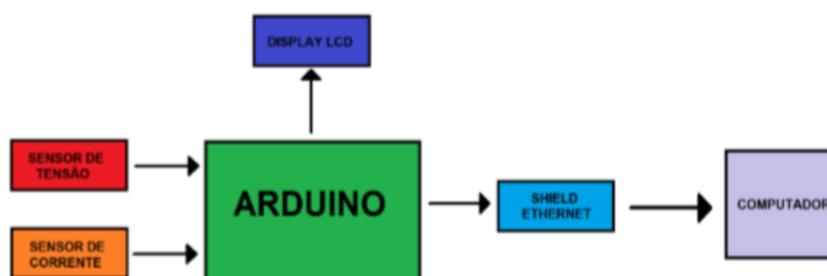
2.8.2 Medidor De Demanda De Energia Elétrica Residencial Com Acesso Remoto

Paula (2013) aborda em sua monografia preocupação que devemos ter em incentivar um consumo mais racional de energia elétrica, principalmente quando se leva em consideração que a maior parte da energia elétrica consumida é originária de fontes não renováveis. E a criação de dispositivos que auxiliem as pessoas a terem um maior conhecimento sobre seu consumo de energia elétrica contribuir para um consumo mais sustentável e ajudar a reduzir impactos ambientais.

O projeto de Paula (2013) teve como objetivo o desenvolvimento de um medidor eletrônico que possibilita a leitura dos dados obtidos remotamente. O dispositivo mede tensão, corrente, potência ativa e reativa, fator de potência e energia.

O autor utilizou para o desenvolvimento do dispositivo proposto o *kit* de prototipagem Arduino MEGA para o processamento dos dados, o sensor de corrente ACS172 e *display* LCD para leitura local dos dados. Também foi utilizado o *shield* ethernet para o compartilhamento dos dados e disponibilização dos mesmos em uma página WEB. O diagrama de funcionamento é mostrado na figura 6.

Figura 6 – Diagrama de funcionamento do projeto



Fonte: (PAULA, 2013)

2.8.3 Sistema Supervisório Para Acompanhamento De Consumo De Energia Elétrica Residencial

Na monografia de Gomes, Gomes e Tagliari (2016), os autores apresentam preocupações acerca do alto custo de produção energia elétrica no Brasil que afetam diretamente os gastos dos consumidores, visto que eventuais aumentos no

custo de produção ao invariavelmente repassados aos usuários. Sendo assim, uma melhor gestão do consumo de energia elétrica torna-se uma interessante alternativa para fins de economia e redução de gastos.

Observando a pouca disponibilização de aplicativos em *software* livre que possibilitem ao consumidor medir e avaliar o consumo de energia elétrica em sua residência, os autores tiveram como objetivo desenvolver sistema supervisorio para acompanhamento de consumo de energia elétrica em uma residência, para que os usuários possam ter um maior controle do seu consumo de energia elétrica.

Para o desenvolvimento do dispositivo os autores utilizaram o Arduino MEGA, o sensor de corrente não invasivo SCT013-030. Os dados obtidos pelo sensor e processados no microcontrolador do Arduino MEGA são armazenados num cartão de memória integrado ao Arduino e então, utilizando o módulo Wi-Fi, também integrado ao Arduino, os dados são enviados a um servidor WEB e disponibilizados para a leitura no aplicativo desenvolvidos pelos autores.

3 MATERIAIS E MÉTODOS

3.1 DISPOSITIVO PROPOSTO

Para o desenvolvimento deste trabalho, foi criado o projeto de um sistema que realizasse as funções desejadas na concepção do dispositivo proposto, no caso, medição, comunicação e compartilhamento e leitura remota dos dados obtidos. O desenvolvimento desse projeto foi pensado tendo a facilidade de manuseio e de portabilidade como prioridades. Sendo assim, para execução desse projeto, foi escolhido para a realização do aferimento da corrente o sensor de corrente não-invasivo SCT013-000 que é a ferramenta responsável por passar ao processador do dispositivo a informação lida sobre a corrente que está passando por certo eletrodoméstico. Para o processamento dos dados, decidiu-se pela utilização do *kit* de prototipagem Arduino Uno e, para o compartilhamento dos dados, utilizou-se o Ethernet *Shield*, ferramenta auxiliar ao Arduino.

O Ethernet *Shield* funciona como um servidor web, assim os dados processados pelo Arduino ficam disponíveis para leitura pelo usuário que pode acessá-los utilizando o navegador de internet do seu computador ou dispositivo móvel acessando o endereço de IP reservado para a página HTML criada. Essa simples página HTML funciona como uma interface do usuário com o sistema, podendo este ligar (solicitar a realização da medição), desligar (interromper a medição que estiver sendo realizada) e ler os dados de consumo por meio da página desenvolvida.

Neste capítulo, serão apresentados todos os passos para desenvolvimento do dispositivo, assim como testes e montagens, para que no capítulo seguinte seja demonstrado os resultados obtidos com os testes.

3.1.1 Implementação do Sistema

O sistema para monitoramento de consumo de energia elétrica particular foi idealizado para ser um dispositivo portátil que pode ser utilizado para o aferimento da taxa de consumo de um aparelho eletrodoméstico específico individualmente ou de um conjunto de aparelhos. O Hardware do dispositivo desenvolvido é composto por três componentes essenciais: o sensor de corrente não invasivo SCT013-000, o

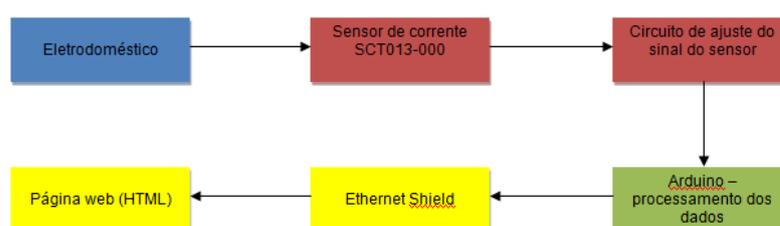
kit de prototipagem Arduino Uno e o Ethernet *Shield*. Além destes, foram utilizados outros componentes para o desenvolvimento do circuito de medição

O sensor de corrente não invasivo SCT013-000 funciona como um alicate amperímetro, possibilitando que a medição seja realizada sem a necessidade de introduzir um sensor no circuito elétrico. O sensor realiza a leitura da corrente do eletrodoméstico instantaneamente e envia os valores para o Arduino. Responsável pelo processamento dos dados, o microcontrolador ATMEGA328, a partir da leitura feita pelo sensor, realiza os cálculos e obtém os resultados da medição realizada.

Através deste sistema, é possível ter conhecimento do consumo de energia elétrica instantaneamente. Isto é facilitado, pois, através do Ethernet *Shield*, que possibilita a conexão com internet, os dados obtidos e processados pelo dispositivo ficam registrados e disponíveis ao usuários imediatamente após a conclusão de uma medição. Isso ocorre porque o Ethernet *Shield* funciona como um servidor web, armazenando os dados e permitindo que os mesmos sejam acessados pelo usuário pouco tempo após a solicitação da interrupção da medição.

O desenvolvimento do sistema foi iniciado com a estruturação do *hardware* que envolve a conexão do sensor de corrente, que realiza a coleta de dados, com o Arduino, que é onde esses dados serão armazenados e processados, e também com o Ethernet *Shield*, que é o responsável pela conexão com a internet para que o compartilhamento dos dados possa ser realizado, tornando as informações obtidas acessíveis ao usuário e possibilitando a leitura das mesmas. O dispositivo ao todo não é muito complexo por contar com *kits* embarcados e tecnologias já condensadas em módulos. Para a alimentação do sistema é necessário uma fonte de alimentação de 5V, podendo essa alimentação ser feita por pilha, bateria, porta USB ou mesmo um carregador de celular com essa tensão de saída. Basicamente o sistema pode ser representado pelo diagrama de blocos da Figura 7.

Figura 7 - Diagrama de Funcionamento do Sistema



Fonte: (Autor)

3.1.2 Hardware

3.1.2.1 Sensor de corrente SCT013-000

Como brevemente apresentado na seção 2.4, o Sensor SCT013-000 é o um sensor corrente que, segundo sua especificação técnica, suporta na entrada uma corrente de até 100A, com saída de 50mA no secundário, funcionando como um transformador, possuindo um fator K de 2000 (que está relacionada às espiras do trafo) e tolerando temperaturas que podem alternar entre -25°C até 70°C . A faixa de operação do sensor é não linear em 3% e possui uma taxa de variação de seu valor nominal que vai se 10% a 120%, equivalente a 10A até 120A.

Pelo fato deste sensor entregar na saída uma variação de corrente, diferentemente dos demais modelos de sensores da família SCT013, que entregam uma variação de tensão na saída, é necessário gerar uma variação de tensão a partir do valor de corrente entregue pelo sensor. Essa variação de tensão é necessária porque o Arduino somente interpreta valores de tensão em suas entradas, assim gerar essa variação de tensão é fundamental para que a leitura da corrente seja feita pelo Arduino. Essa variação pode ser feita através de um resistor de carga, também denominado como “Burden Resistor”.

Para o cálculo do “Burden Resistor”, primeiramente, determina-se a máxima corrente que pode ser medida pelo sensor, que no caso é 100A. Convertendo essa corrente máxima RMS para corrente de pico no primário (4.1):

$$I_{p1} = I_{rsm} \cdot \sqrt{2} = 141,4A \quad (4.1)$$

Com o valor relação de transformação K sendo 2000, número de espiras do trafo, calcula-se a corrente de pico no secundário (4.2):

$$I_{p2} = \frac{141,4}{2000} = 70,7mA \quad (4.2)$$

Para se ter uma melhora na resolução do Arduino, fazendo com que a corrente fique dentro da faixa de operação de tensão deve-se utilizar a metade da tensão máxima do Arduino: 2,5V (4.3).

Portanto:

$$R_{Burden} = \frac{2,5}{0,0707} = 35,4\Omega \quad (4.3)$$

Então de acordo com os cálculos realizados, se um projetista quiser trabalhar com a faixa de leitura do sensor SCT013-000 de 0 a 100A o resistor de carga não pode ser maior que 35,4Ω, senão a tensão na entrada analógica do Arduino poderá não estar dentro da faixa de trabalho do mesmo.

Como para os testes deste projeto as correntes medidas não serão maiores que 20 A, decidiu-se pela utilização de um resistor de carga de 220Ω. Com o resistor de 220Ω, a corrente máxima que pode ser lida é de 16,06A, como mostrado nos cálculos (4.4), (4.5) e (4.6).

$$R_{Burden} = \frac{2,5}{I_{p2}^2} \quad (4.4)$$

$$220 = \frac{2,5}{I_{p2}^2} \quad (4.5)$$

$$I_{p2} = \frac{2,5}{220} = 0,00803A \quad (4.6)$$

De acordo com o cálculo (4.7) realizado, a corrente de pico no secundário não pode ser maior que 0,01136A, sendo a corrente RMS no secundário será no máximo de:

$$I_{rms2} = \frac{0,01136}{1,414} = 0,00803A \quad (4.7)$$

Assim, a corrente RMS máxima no primário é a multiplicação da corrente RMS do secundário pelo fator K (4.8) e (4.9):

$$I_{rms1} = I_{rms2} \cdot K \quad (4.8)$$

$$I_{rms1} = 0,00803 \cdot 2000 = 16,06A \quad (4.9)$$

Concluída a etapa de determinação do resistor de carga, ou resistor Burden, foi necessária a adequação mecânica do sensor SCT013-000 para que este pudesse integrar o dispositivo. O ajuste aqui se deve ao fato de o sensor originalmente apresentar em sua saída um *plug* p2, como pode ser visto na Figura 8, o que impossibilita a conexão deste ao circuito. Sendo assim, foi necessário realizar um corte no cabo do sensor para que se pudesse conectar diretamente o sensor na placa de projeto onde se montou o circuito.

Figura 8 - Sensor de Corrente com *plug* P2



Fonte: (THOMSEN, 2015)

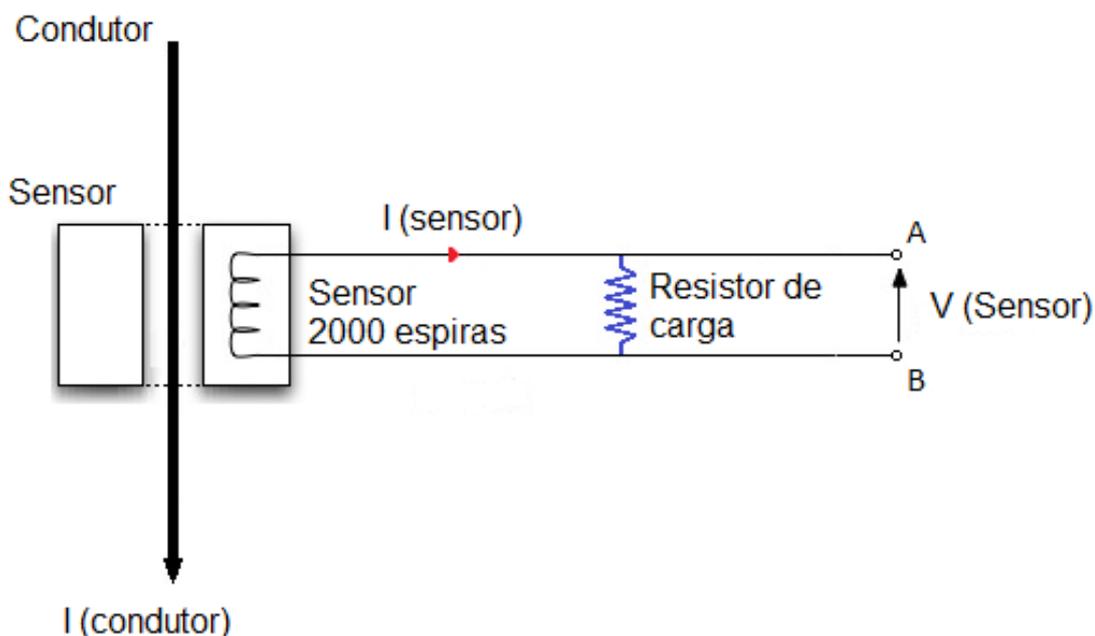
3.1.2.2 O circuito de medição e ajuste do sinal

Como já dito, para que o Arduino possa receber os dados do sensor e realizar o seu processamento, é necessário que se ajuste o sinal entregue pelo sensor. Assim o circuito de ajuste é essencial para o devido funcionamento do projeto.

Um componente fundamental do circuito foi visto anteriormente: o resistor de carga. O resistor de carga, ilustrado na Figura 9, é usado nesse circuito ligando-se aos terminais do sensor de corrente, gerando assim uma variação de tensão. Mas isso não é suficiente para o bom desempenho do sistema. Assim como as entradas

do Arduino não leem tensões além dos 5V, estas também não identificam variações negativas. Com isso, é necessário ajustar a senóide gerada na saída do sensor para que a entrada no Arduino esteja dentro da sua faixa de trabalho, no caso de 0 a 5V.

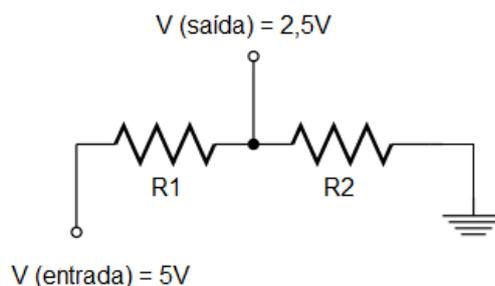
Figura 9 – Esquema do resistor de carga



Fonte: (Autor)

Para adequar a variação de tensão gerada pelo resistor de carga à faixa de trabalho do Arduino une-se ao resistor de carga um simples circuito *offset*, que funciona como uma fonte de tensão somando 2,5V à senóide da tensão gerada pelo resistor de carga. A primeira parte do circuito *offset* é o divisor de tensão, esquematizado na figura 10, que utiliza a própria alimentação de 5V do Arduino. Utiliza-se dois resistores iguais para que a tensão seja igualmente dividida entre eles.

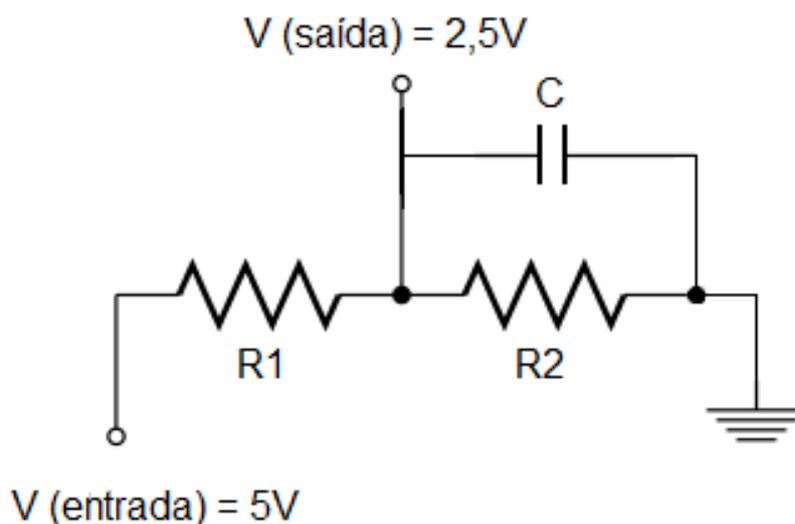
Figura 10 – Divisor de tensão



Fonte: (Autor)

Por fim, adiciona-se um capacitor entre a saída de 2,5V e o terra para finalizar o circuito *offset*, como apresentado na Figura 11.

Figura 11 – Circuito *offset*



Fonte: (Autor)

3.1.2.3 Arduino Uno, Ethernet *Shield* e a comunicação

Concluída a etapa de implementação do circuito responsável pela medição da corrente elétrica, o foco voltou-se para a etapa intermediária do projeto: a fase de comunicação das informações processadas pelo microcontrolador com o servidor web onde os dados medidos pelo dispositivo estarão armazenados e disponíveis para que os usuários possam acessá-los remotamente.

O Arduino Uno e o Ethernet *Shield* já apresentados nas seções 2.3.1 e 2.5, são fundamentais nesta parte do projeto. É por meio deles que o sistema irá realizar o processamento dos seus dados e por fim irá disponibilizar suas informações ao usuário conectando o sistema à internet. É nesta etapa que o software começa a ser desenvolvido.

O Ethernet *Shield* funciona baseado no ethernet *chip* Wiznet W5100. O *Shield* só funciona sendo acoplado ao Arduino. O Ethernet trabalha fornecendo um endereço de IP compatível com os protocolos TCP e UDP gerando um pareamento para transmissão e recepção na rede, possibilitando a conexão do Arduino à internet.

3.1.3 Software

3.1.3.1 Bibliotecas

As bibliotecas utilizadas para o desenvolvimento do código do sistema são mostradas na Figura 12.

Figura 12 – Bibliotecas utilizadas

```
#include <SPI.h>
#include <Ethernet.h>
#include <EmonLib.h>
```

Fonte: (Autor)

A biblioteca SPI.h é essencial para o funcionamento do Arduino. Essa biblioteca consiste em um protocolo de dados seriais síncronos utilizado para a comunicação entre o microcontrolador e os demais periféricos. Assim esta biblioteca é responsável por estabelecer relação de “hierarquia” entre os componentes do circuito, sendo que o geralmente será o “mestre” e os demais periféricos serão os “escravos”.

A biblioteca Ethernet.h tem como principal função disponibilizar todos os protocolos de comunicação para o Ethernet *Shield*, com a finalidade de facilitar realização da comunicação do IP (Protocolo de Internet) em questão com o banco de dados. Mais à frente, serão mostradas linhas do código dando o endereço de usuário para o *shield*.

A EmonLib.h é uma biblioteca desenvolvida pela *Open Energy Monitor* na linha de *Open Source*. A sua função principal no projeto desenvolvido é calcular, dada a entrada do sensor no sistema, o valor de corrente RMS. A Emonlib.h é uma biblioteca que possui variadas funções, tendo no mercado vários dispositivos que a utilizam.

3.1.3.2 Código de comunicação

No trecho de código mostrado na Figura 13, são passados ao Ethernet *Shield* os parâmetros necessários para a comunicação do sistema com a rede. Para que a comunicação ocorra com a correção necessária é preciso que o desenvolvedor

disponibilize um endereço de IP livre na sua rede, bem como também disponibilize ao Ethernet os dados de *gateway* e subrede. A linha de código “EthernetServer Server(80);” é a responsável por determinar a porta que o servidor utilizará para enviar a página web que será criada.

Figura 13 – Parâmetros de comunicação

```
//PARÂMETROS PARA CONEXÃO DO ETHERNET SHIELD
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = { 192, 168, 1, 20 };
byte gateway[] = {192, 168, 1, 1};
byte subnet[] = {255, 255, 255, 0};
EthernetServer server(80);
```

Fonte: (Autor)

Já no trecho de código da Figura 14, as funções mostradas são as responsáveis por inicializar o servidor que fica disponível ao usuário (cliente) sempre que este quiser utilizar o sistema. Quando o usuário fizer uma requisição uma página web será enviada para que ele possa utilizar o sistema.

Figura 14 – Comando para inicialização do servidor

```
void setup(){
  Ethernet.begin(mac, ip, gateway, subnet); //PASSA OS PARÂMETROS PARA A FUNÇÃO QUE VAI FAZER A CONEXÃO COM A REDE
  server.begin(); //INICIA O SERVIDOR PARA RECEBER DADOS NA PORTA 80
```

Fonte: (Autor)

3.1.3.3 Calibração e leitura do sensor

Como dito anteriormente, a biblioteca Emonlib.h é a responsável por realizar os cálculos com as amostras fornecidas pelo sensor. Mas para que esses cálculos sejam precisos é necessário configurar o pino do Arduino para a leitura adequada pelo microcontrolador. Primeiramente é importante que se declare um objeto que será utilizado pelo código para indicação do sensor de corrente, para o código desenvolvido “emon1” é utilizado para representação do sensor no código. E então, o passo seguinte é definir o pino do Arduino com o qual o sensor de corrente estará conectado e passando as amostras obtidas. Nesse projeto decidiu-se pelo pino A1 do Arduino como visto na Figura 15.

Figura 15 – Configuração do sensor no Arduino

```
EnergyMonitor emon1;

//PINO DO SENSOR DE CORRENTE
int pino_sct = A1;
```

Fonte: (Autor)

Para o cálculo da corrente eficaz é utilizada a função “`emon1.current(pin_sct, 9.09090909);`”, como mostrado na Figura 16. A função realiza o cálculo tendo como parâmetro as amostras obtidas pelo pino A1 que as recebe do sensor de corrente. O outro parâmetro (9.09090909) é o valor de calibração. Este valor pode ser alterado de acordo com o resistor de carga utilizado no projeto e pode ser obtido dividindo o fator K do sensor (2000) pelo valor do resistor de carga, no caso do utilizado no projeto (220 Ω).

Figura 16 – Calibração da entrada do Arduino

```
emon1.current(pino_sct, 9.09090909);
```

Fonte: (Autor)

3.1.3.4 Programa para cálculo do consumo

Antes de ser elaborado o programa completo com todas as funções do sistema, optou-se pelo desenvolvimento de um código mais simples que contemplasse somente a verificação da medição de corrente, da potência instantânea e disponibilizasse uma estimativa do consumo. O trecho de código apresentado na Figura 17 indica o cálculo do consumo e da corrente média aferida.

Figura 17 – Cálculo do consumo de energia

```
if (status == 1){
  Irms = emon1.calcIrms(1480);
  if (Irms*127 > 4){
    tempo = millis();
    if((tempo - ult_tempo) >= 1000){
      ult_tempo = tempo;
      seg = seg + 1;
      I = I + Irms;
      cont = cont + 1;
      //MOSTRA QUE O SISTEMA FOI ACIONADO NO MONITOR SERIAL
      Serial.print("\n CONSUMO SENDO AFERIDO \n");
      //MOSTRA A CORRENTE MEDIDA PELO SENSOR NO MONITOR SERIAL
      Serial.print(Irms);
      Serial.print("\n");
    }
    if(seg >= 60){
      seg =0;
      minuto = minuto + 1;
      Serial.print(minuto);
    }
    Imed = I/cont;
    minutos = minuto + seg/60;
  }
}
client.println(Imed*127*1.2315*minutos/60000); //ESCREVE O VALOR DO CONSUMO ESTIMADO
```

Fonte: (Autor)

No capítulo 4, apresentam-se os valores obtidos com os testes que foram realizados.

3.1.3.5 Página web

Para a supervisão do sistema (acionamento e leitura dos dados processados) optou-se pelo desenvolvimento de uma página HTML com um *layout* bem simples, mas que apresentasse de forma didática o status do sistema e seu funcionamento ao usuário. O *layout* mais simples também foi uma alternativa, pois uma página mais recursos estéticos e funcionais pode prejudicar o desempenho do Arduino, pois exigiriam quase 100% da memória do Arduino (ou este não teria memória suficiente para executar uma página mais complexa), o que tende a causar problemas de estabilidade, com desempenho irregular do controlador.

As páginas desenvolvidas apresentam uma estrutura simples com o cabeçalho mostrando o nome do autor e o título do projeto, um botão para ligar ou desligar o sistema e mais algumas linhas de texto para indicar o status do sistema, se o mesmo está ligado ou desligado, bem como para apresentar ao usuário o resultado da medição realizada. O texto HTML da *interface* inicial do projeto é apresentado na Figura 18 e na Figura 19 é mostrado como a página fica visível ao usuário.

Figura 18 – Texto HTML da página

```
<html>
<head>
<link rel='icon' type='image/png' href='raio.jpg'/>
<title>TCC - CONSUMO DE ENERGIA</title>
</head>
<body style='background-color:#202421'>
<center><font color='red'><h1>WELMER SALES DE SOUSA</font></center></h1>
<h1><center>TCC - CONSUMO DE ENERGIA</center></h1>
<center><form method=get name=SISTEMA><input type=hidden name=sistema value=1 /><input type=submit value=INICIAR></form></center>
<center><font size='5'>Status</center>
<center><font color='red' size='5'>DESLIGADO</center>
</body>
</html>
```

Fonte: (Autor)

Figura 19 – Página projetada para o Sistema



Fonte: (Autor)

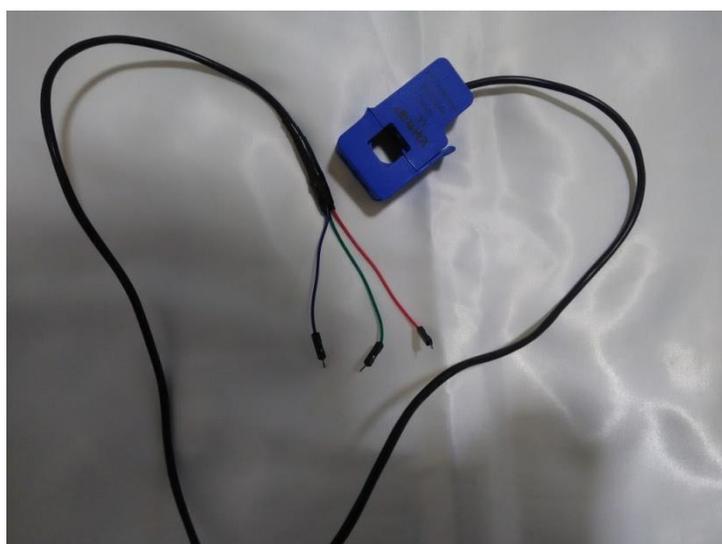
4 RESULTADOS

Após um longo trabalho de pesquisa e estudo sobre a proposta do projeto e sobre as tecnologias envolvidas no mesmo, realizou-se a construção do dispositivo projetado, tendo sido o sistema desenvolvido devidamente testado e alcançando resultados satisfatórios que serão mostrados no decorrer deste capítulo.

4.1 CIRCUITO DE MEDIÇÃO

O desenvolvimento do projeto foi iniciado a partir da criação do circuito eletrônico e do código relacionado à obtenção de informações da rede elétrica, no caso, a medição de corrente do aparelho que está sendo medido. Não houve grandes dificuldades no desenvolvimento desta fase do projeto. De início, foi observado, como citado na seção 4.1.2.1, que seria necessário fazer uma adaptação na saída do sensor de corrente, visto que este originalmente apresenta um *plug* p2 em sua saída. Assim, o cabo do sensor foi cortado e em sua extremidade foram ligados de jumpers que facilitassem a ligação do sensor à placa onde se desenvolveu o circuito de ajuste do sinal. A Figura 20 mostra o sensor modificado.

Figura 20 – Sensor de corrente com sua saída adaptada



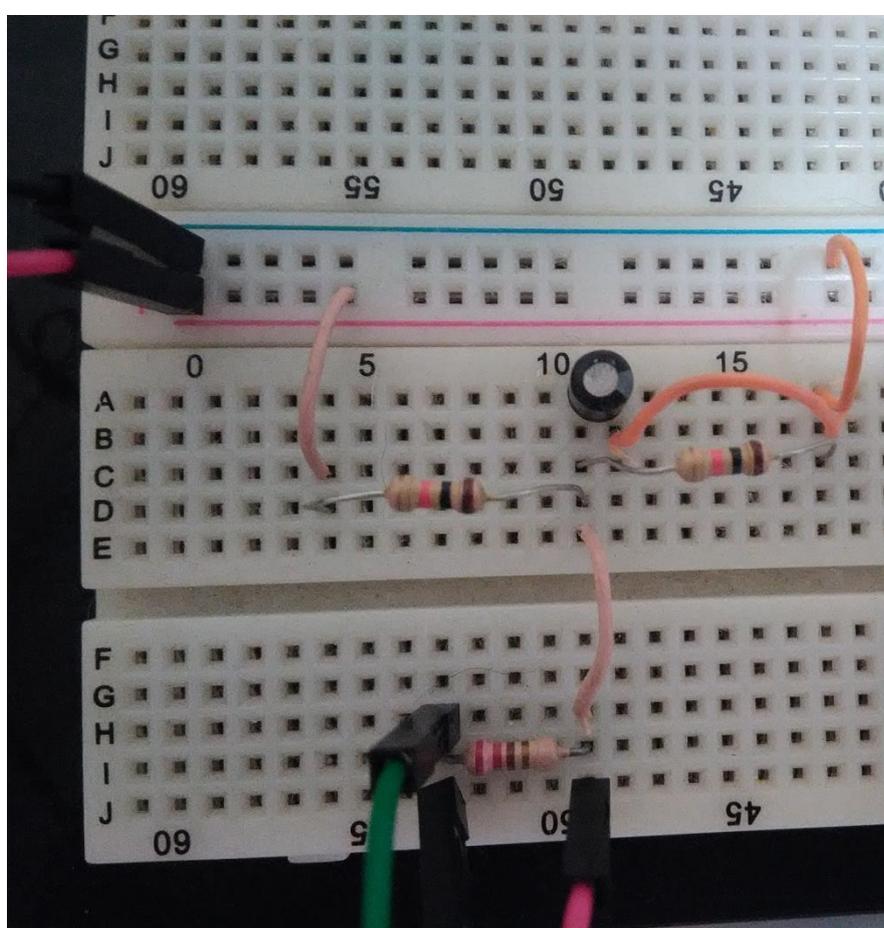
Fonte: (Autor)

Feita a adaptação, o circuito de ajuste do sinal foi desenvolvido. A primeira preocupação aqui foi quanto ao resistor de carga e a faixa de leitura do sistema. Se optasse por um resistor de carga de 33Ω que possibilitaria a faixa de leitura máxima

do sensor de até 100A, o sistema seria pouco sensível a correntes pequenas e a variações leves na corrente que é o que se encontra mais comumente nas instalações residenciais, prejudicando assim a precisão desejada para o sistema. Então, observados esses detalhes, optou-se pela utilização de um resistor de carga de 220Ω , que proporciona uma faixa de leitura de um pouco mais de 16A.

Definido o resistor de carga, o circuito de ajuste de sinal, mostrado na figura 21, com o *offset*, que adapta o sinal enviado pelo sensor à faixa de trabalho do Arduino Uno, foi então concluído.

Figura 21 – Circuito de medição montado

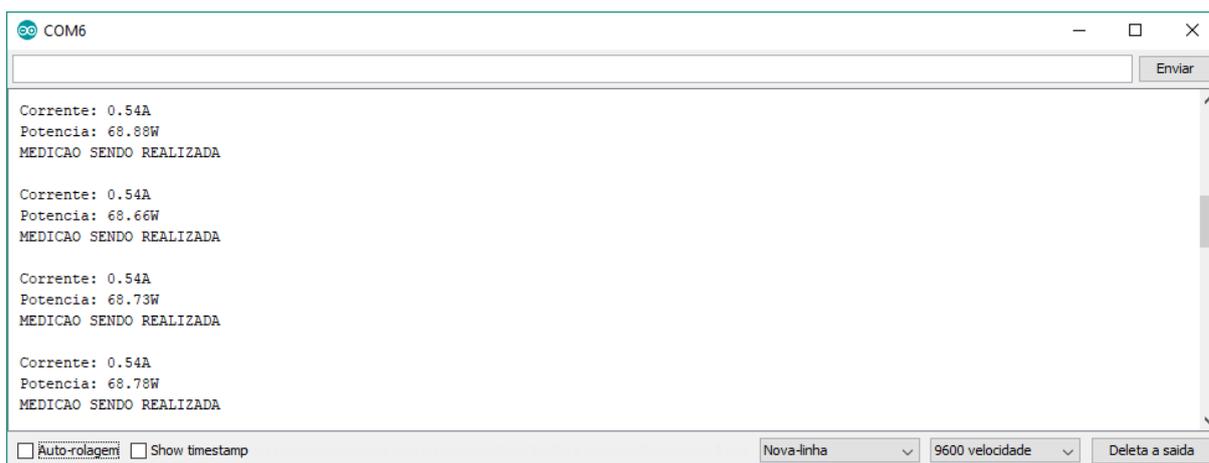


Fonte: (Autor)

Para realização do teste do sensor e do circuito montado, foi desenvolvido um código inicial para observação da leitura feita pelo sensor de corrente. O código imprime na tela do monitor serial do Arduino os valores de corrente RMS obtida pelo sensor e calcula potência instantânea consumida pelo aparelho que está sendo medido. Para fins de verificação e validação dos valores obtidos pelo circuito, os resultados foram comparados com valores de corrente medidos por um amperímetro

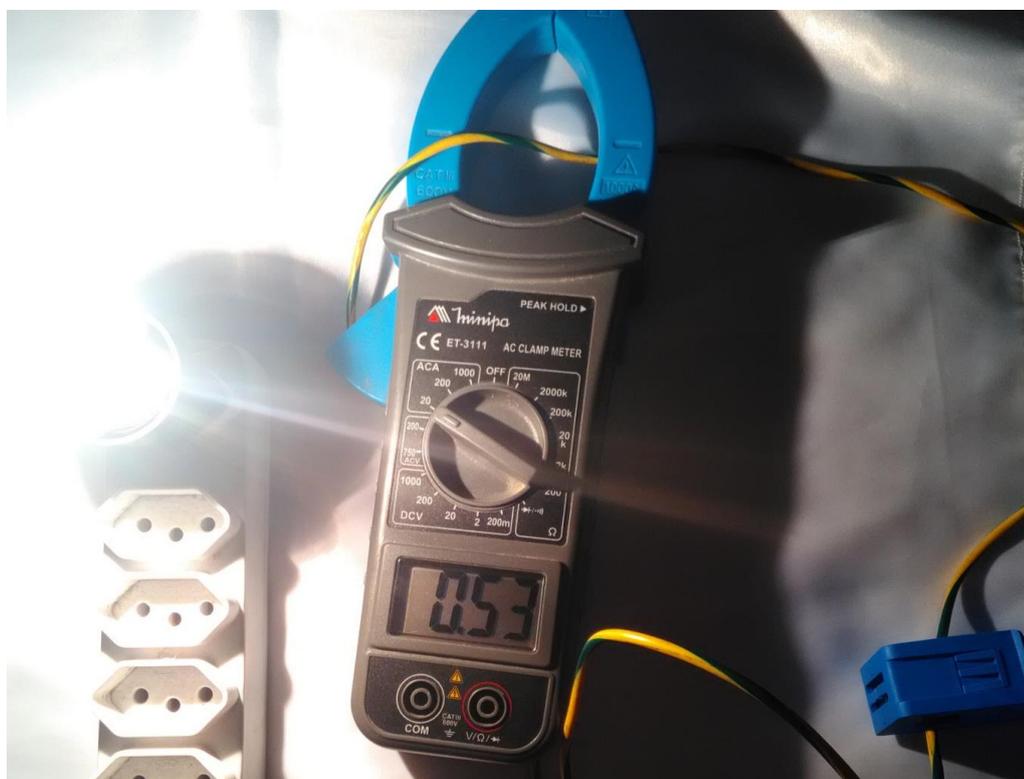
alicate, por este apresentar o mesmo princípio de funcionamento do sensor utilizado no desenvolvimento do projeto. No teste realizado, a corrente e a potência são aferidas a cada um segundo. As figuras 22, 23, 24, 25, 26 e 27 mostram os resultados obtidos pelo sensor e o comparativo dos valores medidos pelo sensor e pelo amperímetro alicate.

Figura 22 – Resultado da medição de corrente no monitor serial (lâmpada 70W)



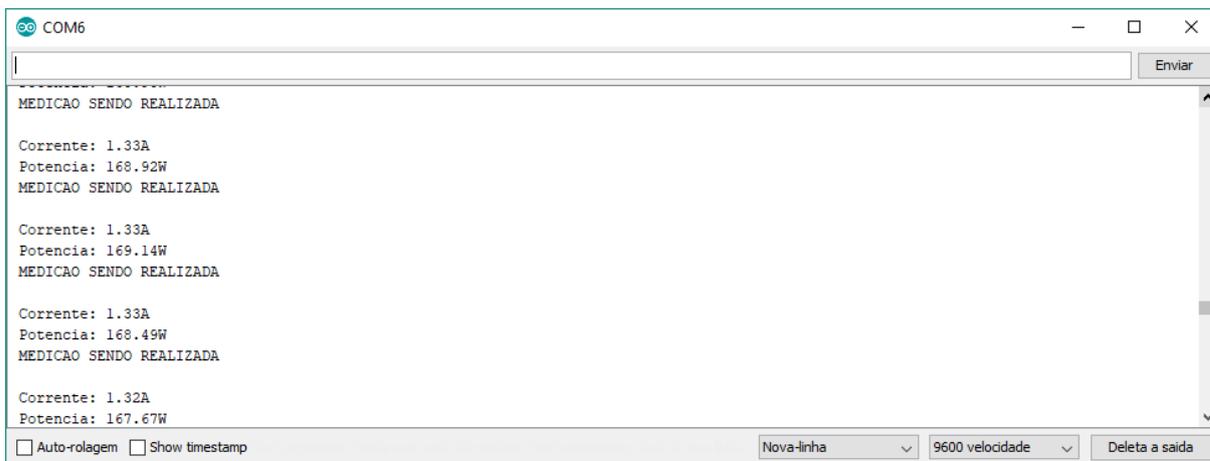
Fonte: (Autor)

Figura 23 – Medição de corrente do Amperímetro (lâmpada 70W)



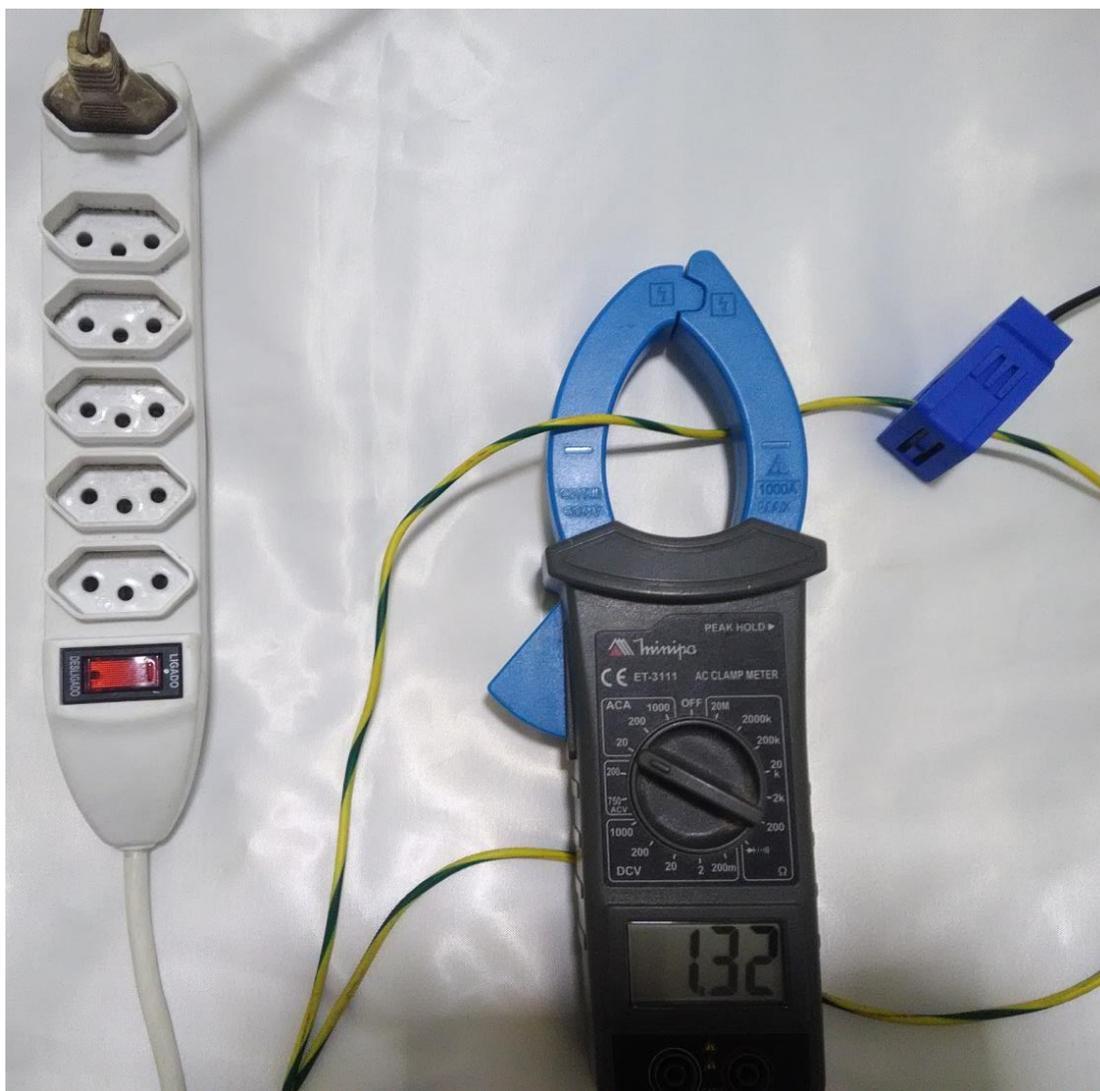
Fonte: (Autor)

Figura 24 – Resultado da medição de corrente no monitor serial (Ventilador)



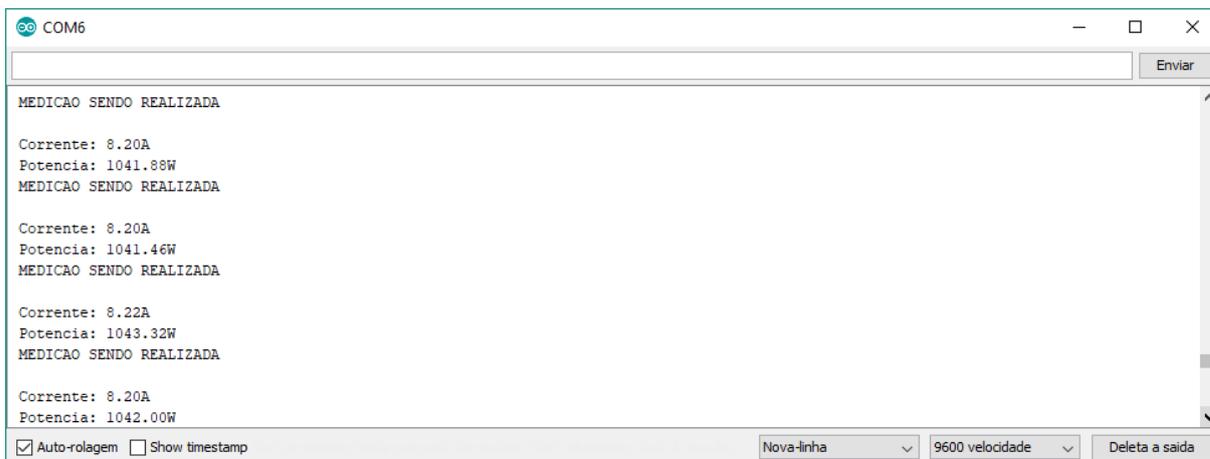
Fonte: (Autor)

Figura 25 – Medição de corrente do Amperímetro (Ventilador)



Fonte: (Autor)

Figura 26 – Resultado da medição de corrente no monitor serial (Ferro de passar)



Fonte: (Autor)

Figura 27 – Medição de corrente do Amperímetro (Ferro de Passar)



Fonte: (Autor)

Como pode ser observado, o valor de corrente medido pelo sistema e o valor aferido pelo amperímetro são bem próximos, podendo considerar satisfatórios os resultados obtidos e validando o funcionamento do sensor e do circuito desenvolvido. Os valores medidos pelo sensor quando em comparação com os

valores obtidos através do amperímetro apresentam-se dentro de uma margem de erro de 5%, conforme mostrado no Quadro 1.

Quadro 1 – Comparativo entre os valores obtidos pelo sensor e pelo amperímetro

Lâmpada 70W			Ventilador			Ferro de Passar		
Amperímetro	Sensor	Erro	Amperímetro	Sensor	Erro	Amperímetro	Sensor	Erro
0,53 ^a	0,54A	1,8%	1,32A	1,33 ^a	0,7%	8,20A	8,22A	0,2%

4.2 COMUNICAÇÃO DO SISTEMA

Realizados os testes com o sensor de corrente e verificada a correção de seu funcionamento, deu-se sequência ao desenvolvimento do projeto focando na parte de comunicação do mesmo. Aqui, o projeto voltou-se mais para o *software*, requerendo maior atenção na programação.

Para verificação do funcionamento do Ethernet *Shield*, testou um código simples, onde quando o usuário (cliente) acessasse o endereço de IP reservado ao Ethernet *Shield* seria aberto no navegador uma simples mensagem dizendo “Você está conectado!”. A Figura 28 apresenta o teste do Ethernet *Shield*.

Figura 28 – Mensagem de teste do Ethernet *Shield*



Fonte: (Autor)

Com a verificação do bom funcionamento do Ethernet *Shield*, deu-se início ao projeto do desenvolvimento da página HTML que é apresentada ao usuário do sistema. Primeiramente, projetou-se uma página mais completa que apresentasse de forma mais didática as funções que o sistema oferece ao usuário. Mas foram encontrados problemas que dificultaram a implementação dessa página. Por ser uma página com mais linhas de código, o programa pro Arduino ficou muito grande,

HTML escrito no Arduino ficou mais leve, facilitando o desempenho do microcontrolador e carregando com mais facilidade a página para o usuário. Assim para efeitos de teste do funcionamento do sistema a página apresentada ao usuário contém menos informações, como mostrado na figura 31.

Figura 31 – Página do sistema

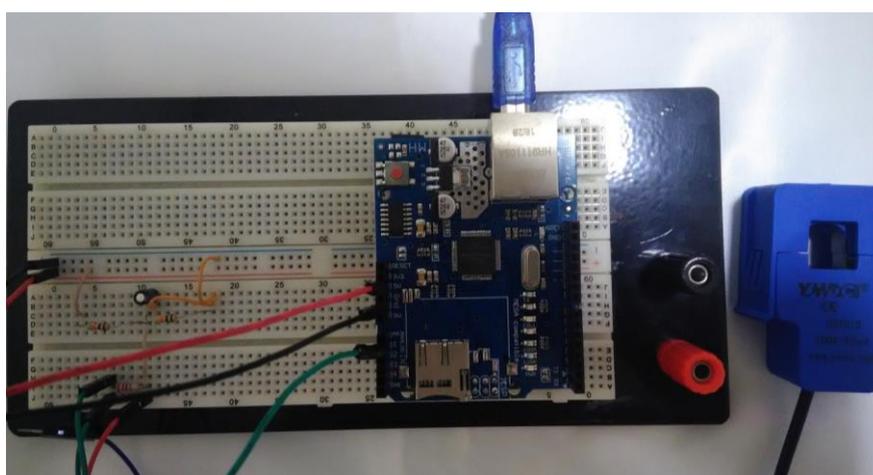


Fonte: (Autor)

4.3 O SISTEMA EM FUNCIONAMENTO

Após a realização dos testes de cada componente fundamental do projeto, foram juntadas as partes que compõem o sistema: o circuito de medição e a parte de comunicação do sistema, o microcontrolador, que compõe a placa do Arduino, e o Ethernet *Shield*, responsável por disponibilizar a página que faz a supervisão do sistema na rede. A figura 32 mostra o circuito completo do sistema.

Figura 32 – Circuito completo do sistema

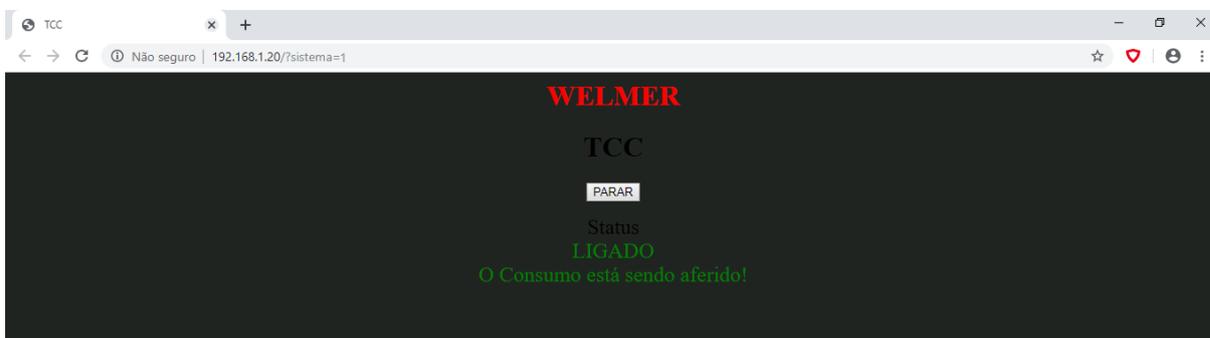


Fonte: (Autor)

Finalizado o desenvolvimento do *Hardware* e concluída a programação do *software* do projeto, foram realizados os testes com sistema completo. Foram

analisadas a comunicação, observando o carregamento da página para o usuário e a funcionalidade das requisições do usuário para o sistema (liga e desligar), e os resultados de medição do sistema que foram apresentados ao usuário durante os testes realizados. Os testes foram realizados na residência do autor deste trabalho, numa rede elétrica monofásica de 127V, aferindo o consumo de três dispositivos. Os aparelhos medidos foram uma sanduicheira, um ferro de passar roupas e um ventilador. Para efeito de comparação, novamente foi utilizado um amperímetro alicate e o tempo de medição foi aferido paralelamente ao sistema para verificação e validação dos resultados obtidos. As figuras 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43 e 44 mostram o sistema em funcionamento e valores aferidos pelo mesmo, bem como a medição realizada pelo amperímetro. Uma observação é que os primeiros testes foram feitos com o plano de fundo da página escuro e durante os testes verificou-se que a leitura da página não era ideal, então para os testes seguintes o plano de fundo foi alterado para uma cor mais clara que permitisse uma leitura mais clara dos dados.

Figura 33 – Página indicando que o sistema está ligado



Fonte: (Autor)

Figura 34 – Página mostrando resultado de medição (Sanduicheira primeiro teste)



Fonte: (Autor)

Figura 35 – Medição da corrente da sanduicheira pelo amperímetro



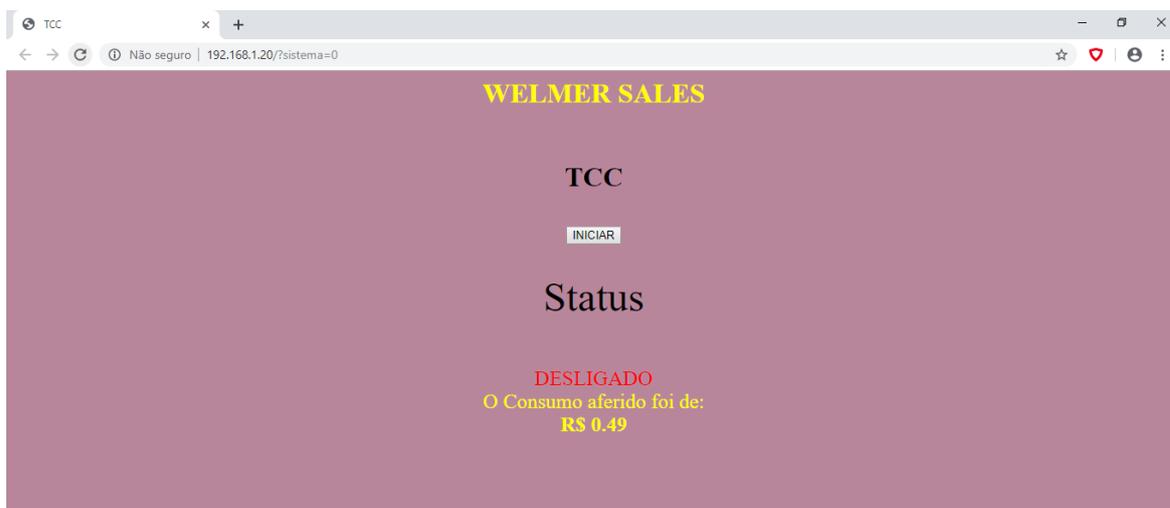
Fonte: (Autor)

Figura 36 – Página mostrando resultado de medição (Sanduicheira – segundo teste)



Fonte: (Autor)

Figura 37 – Resultado de medição (Ferro de passar – primeiro teste)



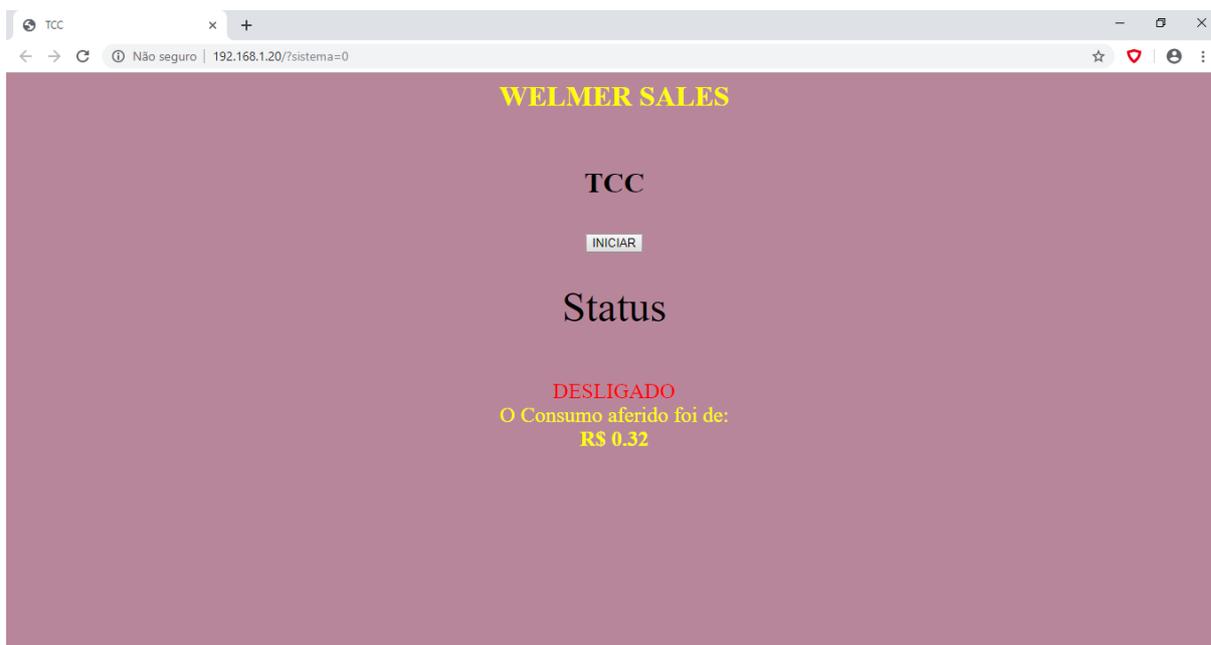
Fonte: (Autor)

Figura 38 – Medição da corrente do ferro de passar roupa pelo amperímetro (primeiro teste)



Fonte: (Autor)

Figura 39 – Resultado de medição (Ferro de passar – segundo teste)



Fonte: (Autor)

Figura 40 – Medição da corrente do ferro de passar roupa pelo amperímetro (segundo teste)



Fonte: (Autor)

Figura 41 – Resultado de medição (Ventilador – primeiro teste)



Fonte: (Autor)

Figura 42 – Medição da corrente do ventilador pelo amperímetro (primeiro teste)



Fonte: (Autor)

Figura 43 – Resultado de medição (Ventilador – segundo teste)



Fonte: (Autor)

Figura 44 – Medição da corrente do ventilador pelo amperímetro (segundo teste)



Fonte: (Autor)

Observou-se com os testes realizados que ainda há certa instabilidade no estabelecimento da comunicação entre o usuário e o servidor, no caso o *Ethernet Shield*, mas essa irregularidade não compromete o funcionamento do sistema. Quando o usuário solicita que o sistema seja ligado, a requisição é atendida, atualizando a página e mostrando ao mesmo que o sistema está aferindo o consumo. Quando ocorre a solicitação para que o sistema seja desligado, a página também é atualizada e imprime para o usuário o valor do consumo aferido como já mostrado pelas figuras. Quanto aos resultados das medições feitas pelo sistema, considerou-se os resultados satisfatórios visto que estes não apresentam grandes

distorções em relação aos valores estimados por cálculo utilizando como parâmetro a corrente medida pelo amperímetro apresentando-se sempre dentro de uma margem de erro de 5%. O Quadro 2 mostra um comparativo entre os resultados medidos e calculados.

Quadro 2 – Comparativo entre os resultados do sistema e o estimado por cálculo

Eletrodoméstico	Corrente Amperímetro	Tempo de Medição	Consumo Calculado	Consumo Medido	Taxa de Erro
Ventilador	1,27A	8h	R\$1,21	R\$1,17	3,4%
Ventilador	1,23A	4h	R\$0,59	R\$0,57	3,5%
Ferro de passar	8,29A	30min	R\$0,50	R\$0,49	2%
Ferro de passar	8,40A	20min	R\$0,33	R\$0,32	3,1%
Sanduicheira	4,30A	15min	R\$0,13	R\$0,13	0%
Sanduicheira	4,30A	25min	R\$0,21	R\$0,20	5%

4.4 CUSTOS PARA O DESENVOLVIMENTO DO PROJETO

Os custos com o desenvolvimento do projeto podem ser observados no Quadro 3, onde estão listados os materiais utilizados e seus respectivos preços.

Quadro 3 - Custo do projeto

Componente	Preço
Arduino Uno	R\$ 50,00
Sensor de Corrente SCT013-000	R\$ 47,00
Ethernet <i>Shield</i>	R\$ 55,00
Jumpers	R\$ 15,00
Outros componentes	R\$ 10,00
	Total = R\$ 177,00

5 CONCLUSÃO

Desde a idealização deste projeto, uma das maiores preocupações que motivou o desenvolvimento do mesmo foi a necessidade de uma maior conscientização a respeito do consumo de energia elétrica a fim de que se tenha um consumo mais sustentável. A situação energética, não só no Brasil, mas em todo o mundo, é uma questão que merece muita atenção.

Ano após ano a preocupação a respeito de mudanças climáticas que podem causar grandes impactos ambientais só aumenta e isso pode atingir diretamente a geração de energia elétrica. O consumo de energia elétrica tem se intensificado com o tempo, e o crescimento populacional juntamente e o surgimento de novos eletrodomésticos, que são incorporados à vida cotidiana, são fatores que contribuem para o aumento da demanda por eletricidade. Juntando isso ao fato de que a geração de energia ainda é majoritariamente feita através da exploração de recursos não renováveis, a escassez desses recursos será inevitável. Com isso, é cada vez mais importante que se busque alternativas para tentar evitar que se chegue nessa situação extrema, seja com a geração de energia renovável ou através de um consumo mais consciente e sustentável.

Tendo essa problemática em mente o projeto foi idealizado e com o desenvolvimento do mesmo, vislumbrou-se a possibilidade de tornar o dispositivo construído em uma ferramenta eficiente e acessível à sociedade com o objetivo não só de proporcionar o controle de gastos, mas também como ferramenta para que os usuários possam diminuir o desperdício e contribuir para um mundo mais sustentável. Os testes realizados foram capazes de constatar que, embora o sistema tenha apresentado certa imprecisão na medição do tempo de consumo, este oferece uma estimativa de consumo realista e que pode proporcionar aos usuários um maior controle dos seus gastos. Assim, podendo ser um instrumento eficiente na conscientização dos seus usuários.

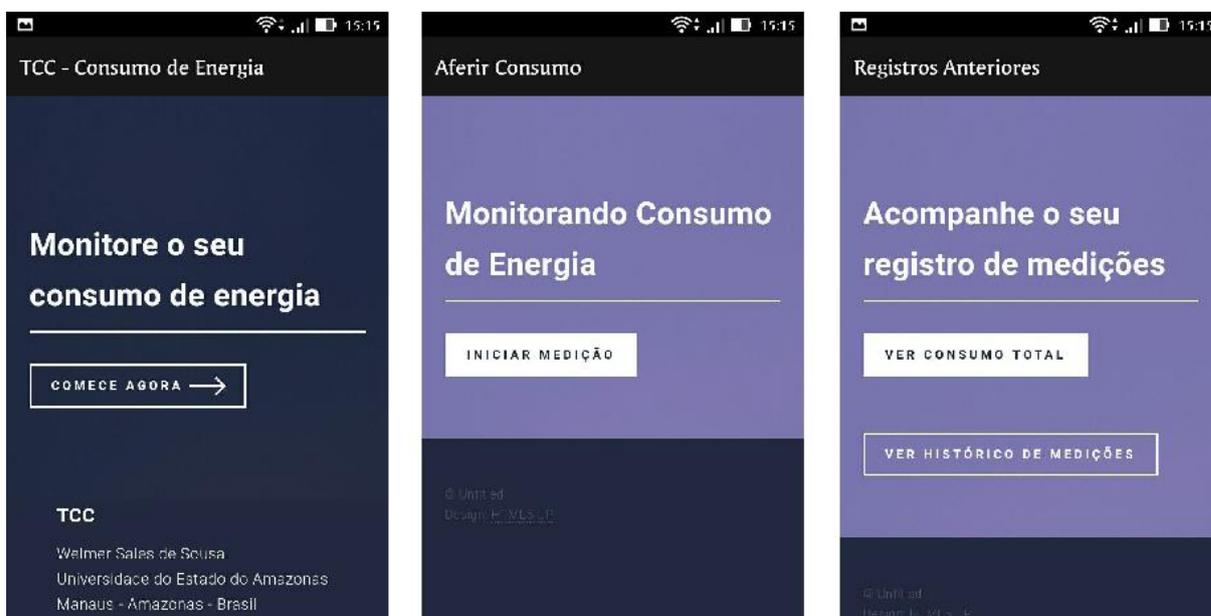
Por fim, muito foi ponderado sobre a preocupação a situação energética do planeta e embora alternativas sejam apresentadas de tempos em tempos, acredita-se que o mais eficiente no momento é conscientizar cada vez mais as pessoas sobre o consumo de energia. Com uma população mais consciente e preocupada com o impacto de suas ações no meio ambiente, poder-se-á oferecer às gerações futuras um mundo mais sustentável. E como já dito, oferecer uma ferramenta que

auxilie a conscientização da sociedade foi uma das principais justificativas desse projeto. Então, levando-se em conta os resultados obtidos, este trabalho contribuirá não só como estudo científico para outros projetos e pesquisas, mas também para que o usuário possua mais informação a respeito do seu consumo de energia elétrica e que, a partir disso, possa tomar ações a fim de se ter um consumo mais sustentável.

5.1 DIFICULDADES ENCONTRADAS

Durante o desenvolvimento do projeto algumas dificuldades foram enfrentadas, as principais estão relacionadas à implementação do *software* do projeto. A primeira dificuldade encontrada está ligada à página de supervisão do projeto. A intenção inicial era apresentar uma interface mais completa, que disponibilizasse mais informações ao usuário, sobretudo de medições anteriormente realizadas, como mostrada na Figura 45. Mas por conta da capacidade de memória do Arduino e da queda de desempenho do sistema, não foi possível apresentá-la, optando assim por uma versão mais simples da página.

Figura 45 – Interface projetada para o sistema



Fonte: (Autor)

A outra grande dificuldade enfrentada tem relação com a medição de tempo do sistema. Embora os resultados apresentados sejam satisfatórios, o sistema

apresenta uma pequena distorção na medição do tempo, que pode comprometer resultados para aferições muito longas.

5.2 TRABALHOS FUTUROS

Neste tópico, são apresentadas algumas ideias de trabalhos futuros, assim como possibilidades de aprimoramento do sistema desenvolvido.

- Medição do tempo: como citado anteriormente o algoritmo para o aferimento do tempo no sistema pode ser otimizado tornando o sistema ainda mais preciso em sua medição;
- Utilização de internet móvel para compartilhamento dos dados: outra possibilidade para melhoria do sistema é a possibilidade de uso de internet móvel para o compartilhamento de dados, podendo as atualizações sobre o fornecimento de energia em sua residência ser disponibilizadas ao usuário mesmo com uma eventual queda de energia;
- Aperfeiçoamento da página de supervisão do sistema: com um novo método de compartilhamento de dados é possível aprimorar a apresentação dos dados ao usuário, podendo ser desenvolvido um site que ofereça mais recursos a quem esteja utilizando o sistema.

REFERÊNCIAS BIBLIOGRÁFICAS

AZEVEDO, Philipe Kretzer e Castro de. **Desenvolvimento de um sistema supervisorio e lógicas de CLP no ambiente de Geração de Energia**. 2013. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia de Controle e Automação, Departamento de Automação e Sistemas, Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis, 2013. Disponível em: <https://repositorio.ufsc.br/bitstream/handle/123456789/169712/PFC_20131-Philipe%20Kretzer.pdf?sequence=1>. Acesso em 4 de abril de 2018.

BARRETO, Guilherme Pimenta. **Sistema de monitoramento e controle de consumo de energia elétrica para uma residência**. 2015. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia de Computação, Faculdade de Tecnologia e Ciências Sociais Aplicadas, Centro Universitário de Brasília, Brasília, 2015. Disponível em: <<http://repositorio.uniceub.br/bitstream/235/7579/1/21136802.pdf>>. Acesso em: 3 de abril de 2018.

BRITO, José Luis Grizinsky de. **Sistema para monitoramento de consumo de energia elétrica particular, em tempo real e não invasivo utilizando a tecnologia Arduino**. 2016. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia Elétrica, Departamento de Engenharia Elétrica, Centro de Tecnologia e Urbanismo, Universidade Estadual de Londrina, Londrina, 2016. Disponível em: <http://www.uel.br/ctu/deel/TCC/TCC2016_JoaoLuisGrizinskyBrito.pdf>. Acesso em: 3 de abril de 2018.

GOMES, Christine Helene; GOMES, Cynthia Criswall Mendonça; TAGLIARI, Simone De Lima. **Sistema supervisorio para acompanhamento de consumo de energia elétrica residencial**. 2016. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia Elétrica, Departamento Acadêmico de Eletrotécnica, Universidade Tecnológica Federal do Paraná, Curitiba, 2016. Disponível em: <http://nupet.daelt.ct.utfpr.edu.br/tcc/engenharia/doc-equipe/2015_1_04/2015_1_04_final.pdf>. Acesso em: 4 de abril de 2018.

KRUTZ, Ronald L. **Securing Scada Systems**. Indianapolis: Wiley Publishing, 2006.

MATOS, Beatriz Rezener Dourado; SILVA, João Gabriel de Britto e. **Estudo comparativo entre o desenvolvimento de aplicativos móveis utilizando abordagem nativa e multiplataforma**. 2016. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia de Software, Faculdade UnB

Gama, Universidade de Brasília, Brasília, 2016. Disponível em: <<https://fga.unb.br/articles/0001/6817/tcc.pdf>>. Acesso em: 5 de abril de 2018.

MCROBERTS, Michael. **Arduino Básico**. 1ª Edição. São Paulo: Novatec Editora, 2011.

MORAES, Daniel de; SILVA, Maurício Tomim Saraiva da; ARRUDA, Rodrigo de Andrade. **Desenvolvimento de um sistema de comunicação em rede profibus entre clp's siemens s7 300 e implementação de sistema supervisorio**. 2017. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia de Controle e Automação, Departamento Acadêmico de Eletrotécnica, Universidade Tecnológica do Paraná, Curitiba, 2017. Disponível em: <http://nupet.daelt.ct.utfpr.edu.br/tcc/engenharia/doc-equipe/2016_1_33/2016_1_33_final.pdf>. Acesso em: 4 de abril de 2018.

NICOLOSI, Denys E. C. **Microcontrolador 8051 detalhado**. 9ª Edição. São Paulo: Editora Érica, 2013.

PAULA, Gilberto José de. **Medidor de demanda de energia elétrica com acesso remoto**. 2013. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia da Computação, Faculdade de Tecnologia e Ciências Sociais Aplicadas, Centro Universitário de Brasília, Brasília, 2013. Disponível em: <http://www.repositorio.uniceub.br/bitstream/235/3857/1/Gilberto%20Jos%C3%A9%20De%20Paula%20Monografia%201_2013.pdf>. Acesso em: 3 de abril de 2018.

PEREIRA, Vidal. **Microcontroladores PIC 16F e 18F teoria e prática**. São Paulo: Instituto Newton C Braga, 2013.

RESULTADOS PROCEL 2020 - ANO BASE 2019. **PROCELINFO**, 2021. Disponível em: <<http://www.procelinfo.com.br/services/procelinfo/Simuladores/DownloadSimulator.asp?DocumentID=%7BEB47D8A6%2DB6C6%2D42A4%2DBF31%2D746D5952C837%7D&ServiceInstUID=%7B5E202C83%2DF05D%2D4280%2D9004%2D3D59B20BEA4F%7D>>. Acesso em 10 de novembro de 2021.

SILVA, Maurício Samy. **Fundamentos de HTML5 e CSS3**. São Paulo: Novatec, 2015.

SOARES, Marco Aurélio. **Aplicativo móvel para academia: estudo de tecnologias e desenvolvimento**. 2016. Monografia apresentada como pré-requisito para conclusão do curso de Sistemas de Informação, Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais, Sabará, 2016. Disponível em: <<https://www2.ifmg.edu.br/sabara/biblioteca/trabalhos-de-conclusao-de-curso/tcc-documentos/TCCMarcoAurelio.pdf>>. Acesso em: 5 de abril de 2018.

SOUZA, Ana Julia Soares de. **Sistema de sugestões para configurações de servidores web apache**. 2013. Monografia apresentada como pré-requisito para conclusão do curso de Ciência da Computação, Centro Universitário Eurípedes de Marília, Marília, 2013. Disponível em: <<http://aberto.univem.edu.br/bitstream/handle/11077/974/Sistema%20de%20Sugestões%20para%20Configurações%20de%20Servifores%20Web%20Apache.pdf?sequence=1&isAllowed=y>>. Acesso em 5 de abril de 2018.

SOUZA, Andrey Fürst de; NUNES, Glauber Muzyka Oyarzabal; BIANCHINI, Thiago Hoppen. **Sistema supervisorio para monitoramento de consumo de água**. 2013. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia Elétrica, Departamento Acadêmico de Eletrotécnica, Universidade Tecnológica do Paraná, Curitiba, 2016. Disponível em: <http://nupet.daelt.ct.utfpr.edu.br/tcc/engenharia/doc-equipe/2015_2_07/2015_2_07_final.pdf>. Acesso em: 4 de abril de 2018.

THOMSEN, Adilson. filipeflop.com. **Como comunicar com o Arduino Ethernet Shield W5100**, 2014. Disponível em: <<https://www.filipeflop.com/blog/tutorial-ethernet-shield-w5100/>>. Acesso em: 12 de abril de 2019.

THOMSEN, Adilson. filipeflop.com. **O que é Arduino?**, 2014. Disponível em: <<https://www.filipeflop.com/blog/o-que-e-arduino/>>. Acesso em: 28 de abril de 2018.

THOMSEN, Adilson. filipeflop.com. **Medidor de corrente não invasivo com Arduino**, 2015. Disponível em: <<https://www.filipeflop.com/blog/medidor-de-corrente-sct013-com-arduino/>>. Acesso em: 28 de abril de 2018.

VILLARIM, Andréa Willa Rodrigues. **Desenvolvimento de sistema supervisorio para dispositivos computacionais portáteis**. 2016. Monografia apresentada como pré-requisito para conclusão do curso de Engenharia Elétrica, Departamento de Engenharia Elétrica, Centro de Energias Alternativas e Renováveis, Universidade Federal da Paraíba, João Pessoa, 2016. Disponível em: <http://www.cear.ufpb.br/arquivos/cgee/TCC/TCC_-_ANDR%C3%89A_WILLA_RODRIGUES_VILLARIM.pdf>. Acesso em: 5 de abril de 2018.

APÊNDICE A – PROGRAMAÇÃO DO MICROCONTROLADOR

```
//SKETCH TCC
//WELMER SALES DE SOUSA

#include <SPI.h>
#include <Ethernet.h>
#include <EmonLib.h>

//PARÂMETROS PARA CONEXÃO DO ETHERNET SHIELD
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = { 192, 168, 1, 20 };
byte gateway[] = {192, 168, 1, 1};
byte subnet[] = {255, 255, 255, 0};
EthernetServer server(80);

//VARIÁVEL PARA BUSCAR DADOS NO ENDEREÇO (URL)
String readString = String(30);

//DECLARAÇÃO DE VARIÁVEL DO TIPO INTEIRA (SERÁ RESPONSÁVEL POR
VERIFICAR O STATUS ATUAL DO SISTEMA)
int status = 0;

EnergyMonitor emon1;

//PINO DO SENSOR DE CORRENTE
int pino_sct = A1;

//int potenciaacumulada=0, consumototal=0, energiatotal=0, horatotal=0;

static unsigned long ult_tempo = 0;
```

```

float minuto = 0, seg = 0, minutos = 0, l = 0, cont = 0, lmed;
int tempo;
double lrms;

void setup(){
  //passa os parâmetros para a função que vai fazer a conexão com a rede
  Ethernet.begin(mac, ip, gateway, subnet);
  //inicia o servidor para receber dados na porta 80
  server.begin();
  Serial.begin(9600);
  //Pino, calibracao - Cur Const= Ratio/BurdenR. 2000/220 = 9.0909
  emon1.current(pino_sct, 9.09090909);
}

void loop(){
  if (status == 1){
    lrms = emon1.calcLrms(1480);
    if (lrms*127 > 4){
      tempo = millis();
      if((tempo - ult_tempo) >= 1000){
        ult_tempo = tempo;
        seg = seg + 1;
        l = l + lrms;
        cont = cont + 1;
        //mostra que o sistema foi acionado no monitor serial
        Serial.print("\n CONSUMO SENDO AFERIDO \n");
        //mostra a corrente medida pelo sensor no monitor serial
        Serial.print(lrms);
        Serial.print("\n");
      }
    }
  }
}

```

```

}
if(seg >= 60){
    seg =0;
    minuto = minuto + 1;
    Serial.print(minuto);
}
lmed = l/cont;
minutos = minuto + seg/60;
}
}
EthernetClient client = server.available(); //cria uma conexão com o cliente
if (client){ //verifica se existe cliente
    while (client.connected()){//enquanto o cliente estiver conectado, executa
        if (client.available()){ //se o cliente está habilitado, faz
            char c = client.read(); //lê caractere a caractere da requisição http
            if (readString.length() < 100){ //se o array for menor que 100, faz
                readString += c; // "readstring" vai receber os caracteres lido
            }
            if (c == '\n'){ //se encontrar "\n" é o final do cabeçalho da requisição http
                if (readString.indexOf("?") <0){ //se encontrar o caracter "?", faz
                }
                else //senão,faz
                    if (readString.indexOf("sistema=1") > 0){ //se encontrar o parâmetro "sistema=1",
faz
                        status = 1; //variável recebe valor 1 (significa que o sistema está ligado)
                    }
                else{ //senão, faz
                    status = 0; //variável recebe valor 0(significa que o led está desligado)
                }
            }
            client.println("HTTP/1.1 200 OK"); //escreve para o cliente a versão do http

```

```

    client.println("Content-Type: text/html"); //escreve para o cliente o tipo de
conteúdo(texto/html)

    client.println("");

    client.println("<!DOCTYPE HTML>"); //informa ao navegador a especificação do
html

    client.println("<html>"); //ABRE A TAG "html"

    client.println("<head>"); //ABRE A TAG "head"

    client.println("<title>TCC</title>"); //ESCREVE O TEXTO NA PÁGINA

    client.println("</head>"); //FECHA A TAG "head"

    client.println("<body>"); //DEFINE A COR DE FUNDO DA PÁGINA

    client.println("<center><font color='yellow'><h1>TCC - WELMER
SALES</font></center></h1>"); //ESCREVE O TEXTO EM COR AMARELA NA
PÁGINA

    client.println("<br />"); //LINHA EM BRANCO

    //client.println("<h1><center>TCC</center></h1>"); //ESCREVE O TEXTO NA
PÁGINA

    if (status == 1){ //SE VARIÁVEL FOR IGUAL A 1, FAZ

        //A LINHA ABAIXO CRIA UM FORMULÁRIO CONTENDO UMA ENTRADA
INVISÍVEL(hidden) COM O PARÂMETRO DA URL E CRIA UM BOTÃO PARAR
(CASO O SISTEMA ESTEJA LIGADO)

        client.println("<br />"); //LINHA EM BRANCO

        client.println("<center><form method=get name=SISTEMA><input type=hidden
name=sistema value=0 /><input type=submit value=PARAR></form></center>");

    }

    else{ //SENÃO, FAZ

        //A LINHA ABAIXO CRIA UM FORMULÁRIO CONTENDO UMA ENTRADA
INVISÍVEL(hidden) COM O PARÂMETRO DA URL E CRIA UM BOTÃO ACENDER
(CASO O SISTEMA ESTEJA DESLIGADO)

        client.println("<br />"); //LINHA EM BRANCO

        client.println("<center><form method=get name=SISTEMA><input type=hidden
name=sistema value=1 /><input type=submit value=INICIAR></form></center>");

    }

    client.println("<br />"); //LINHA EM BRANCO

```

```

    client.println("<center><font size='8'>Status</center>"); //ESCREVE O TEXTO NA
PÁGINA

    if (status == 1){ //SE VARIÁVEL FOR IGUAL A 1, FAZ

        client.println("<center><font color='green' size='5'>LIGADO</center>");
//ESCREVE "LIGADO" EM COR VERDE NA PÁGINA

    }

    if (status == 0){ //SENÃO, FAZ

        if (minutos > 0) {

            client.println("<br />"); //LINHA EM BRANCO

            client.println("<center><font color='red' size='5'>DESLIGADO</center>");
//ESCREVE "DESLIGADO" EM COR VERMELHA NA PÁGINA

            client.println("<center><font color='yellow' size='5'>O ultimo consumo
aferido:</center>"); //ESCREVE O TEXTO EM AMARELO NA PÁGINA

            client.println("<center><b>"); //CENTRALIZA E COLOCA O TEXTO EM
NEGRITO

            client.println("R$ ");

            client.println(lmed*127*1.2315*minutos/60000); //ESCREVE O VALOR DO
CONSUMO ESTIMADO

            client.println("</b></center>");

            client.println("<center><font color='yellow' size='5'>Corrente media:</center>");
//ESCREVE O TEXTO EM AMARELO NA PÁGINA

            client.println("<center><b>"); //CENTRALIZA E COLOCA O TEXTO EM
NEGRITO

            client.println(lmed); //ESCREVE O VALOR DO CORRENTE MÉDIA

            client.println("A");

            client.println("</b></center>");

            minuto = 0;

            seg = 0;

            tempo = 0;

            ult_tempo = 0;

            l = 0;

            cont = 0;

        }

```

```
else{
    client.println("<center><font color='red' size='5'>DESLIGADO</center>");
    //ESCREVE "DESLIGADO" EM COR VERMELHA NA PÁGINA
}
}

//client.println("<hr/>"); //TAG HTML QUE CRIA UMA LINHA HORIZONTAL NA
PÁGINA

//client.println("<hr/>"); //TAG HTML QUE CRIA UMA LINHA HORIZONTAL NA
PÁGINA

client.println("</body>"); //FECHA A TAG "body"
client.println("</html>"); //FECHA A TAG "html"
readString=""; //A VARIÁVEL É REINICIALIZADA
client.stop(); //FINALIZA A REQUISIÇÃO HTTP E DESCONECTA O CLIENTE
}
}
}
}
}
```