



CENTRO UNIVERSITÁRIO DE BRASÍLIA- UniCEUB
FACULDADE DE TECNOLOGIA E CIÊNCIAS SOCIAIS APLICADAS – FATECS
CURSO DE ENGENHARIA DA COMPUTAÇÃO

MATEUS DE MELO SOBRAL

**SISTEMA DE MONITORAMENTO REMOTO
DE CONSUMO DE ENERGIA E
TEMPERATURA EM RACKS DE
DATACENTER**

BRASÍLIA – DF

2º SEMESTRE DE 2015

MATEUS DE MELO SOBRAL

SISTEMA DE MONITORAMENTO REMOTO DE CONSUMO DE ENERGIA E TEMPERATURA EM RACKS DE DATACENTER

Trabalho apresentado ao Centro Universitário de Brasília (UniCEUB) como pré-requisito para a obtenção de Certificado de Conclusão de Curso de Engenharia de Computação.

Orientador: Prof. MSc. Luciano Henrique Duque

Brasília - DF

Dezembro, 2015

MATEUS DE MELO SOBRAL

SISTEMA DE MONITORAMENTO REMOTO DE CONSUMO DE
ENERGIA E TEMPERATURA EM RACKS DE DATACENTER

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB) como pré-requisito
para a obtenção de Certificado de
Conclusão de Curso de
Engenharia de Computação.
Orientador: Prof. MsC. Luciano
Henrique Duque.

Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro de
Computação, e aprovado em sua forma final pela Faculdade de Tecnologia e Ciências
Sociais Aplicadas -FATECS.

Prof. Abiezer Amarilia Fernandes
Coordenador do Curso

Banca Examinadora:

Prof. Orientador: Luciano Henrique Duque
Mestre, UniCEUB

Prof. Nilo Sérgio Soares Ribeiro
Mestre, UniCEUB

Prof. Miguel Eduardo Ordoña Mosquera
Mestre, UniCEUB

DEDICATÓRIA

Dedico este trabalho à minha família, que me apoiou durante todo o curso de graduação, me educou e me ensinou todos os valores que tenho. Dedico também aos amigos que me apoiaram durante o curso.

Mateus de Melo Sobral

AGRADECIMENTOS

Agradeço à minha família por ter me dado a oportunidade de ter chegado até aqui. Em especial ao meu pai, minha mãe e minha irmã. E ao meu orientador Luciano Henrique Duque, professor mestre do UniCEUB, que contribuiu de forma significativa para o desenvolvimento do meu projeto.

Muito obrigado!

Mateus de Melo Sobral

CITAÇÃO

“O saber a gente aprende com os mestres e os livros. A sabedoria, se aprende é com a vida e os humildes.”

Cora Coralina

SUMÁRIO

CAPÍTULO 1 INTRODUÇÃO	15
1.1 Objetivos do trabalho.....	17
1.2 Motivação.....	17
1.3 Metodologia	18
1.4 Resultados Esperados	19
1.5 Estrutura do trabalho	20
CAPÍTULO 2 REFERENCIAL TEÓRICO.....	21
2.1 Tecnologia da Informação Verde	21
2.2 Consumo de Energia Elétrica.....	22
2.3 Racks de DataCenter	22
2.4 Descrição dos Materiais Utilizados	23
2.4.1 Plataforma Arduino	23
2.4.2 IDE do Arduino	26
2.4.1 Sensor de temperatura	27
2.4.2 Sensor de corrente	30
2.4.3 Ethernet Shield.....	34
CAPÍTULO 3 DESENVOLVIMENTO DO PROTÓTIPO	37
3.1 Apresentação Geral do Sistema Proposto	37
3.2 Estação de coleta de dados.....	40
3.3 Módulo Ethernet.....	44
3.4 Desenvolvimento do Software.....	46
3.5 Rede local	51
3.6 Sensor de tensão.....	53
3.7 Placa do circuito impresso	55
3.8 Protótipo final	57
CAPÍTULO 4 TESTES E RESULTADOS	60
4.1 Cenários de testes	60
4.1.1 Cenário 1	60
4.1.2 Cenário 2	62
4.1.1 Cenário 3	63
4.1.2 Cenário 4	66

4.1.3 Cenário 5 67

CAPÍTULO 5 CONCLUSÃO 69

5.1 SUGESTÕES PARA TRABALHOS FUTUROS 71

REFERÊNCIAS..... 72

LISTA DE FIGURAS

Figura 2-1 – Arduino UNO	24
Figura 2-2 – Esquema elétrico da plataforma Arduino UNO	25
Figura 2-3 – IDE do Arduino	27
Figura 2-4 – Gráfico de constante térmica do tempo	28
Figura 2-5 – Conexões do sensor LM35 com Arduino.....	30
Figura 2-6 – Conexões do sensor ACS712 com Arduino	31
Figura 2-7 – Diagrama do circuito integrado	32
Figura 2-8 – Diagrama do circuito integrado	33
Figura 2-9 – Diagrama do circuito integrado	33
Figura 2-10 – Conexão do Ethernet Shield com o Arduino	35
Figura 3-1 – Fluxograma do Projeto	38
Figura 3-2 – Diagrama do Projeto	38
Figura 3-3 – Circuito do sensor LM35	41
Figura 3-4 – Montagem do sensor LM35	42
Figura 3-5 – Circuito do sensor ACS712	42
Figura 3-6 – Montagem do sensor ACS712	43
Figura 3-7 – Montagem do Ethernet Shield ao Arduino	44
Figura 3-8 – Componentes do Ethernet Shield	45
Figura 3-9 – Tabela com informações de rede do Arduino.....	53
Figura 3-10 – Circuito do sensor de tensão.....	54
Figura 3-11 – Esquemático elétrico do circuito proposto.....	56
Figura 3-12 – Layout da placa de cobre	57
Figura 3-13 – Protótipo final	58
Figura 3-14 – Protótipo final com Roteador de rede	59
Figura 4-1 – Desempenho do sensor LM35.....	61
Figura 4-2 – Teste do LM35 com o Clima Tempo.....	61
Figura 4-3 – Teste do ACS712.....	62
Figura 4-4 – Desempenho do ACS712.....	63
Figura 4-5 – Medição da tensão elétrica pelo sensor desenvolvido e pelo multímetro.....	64
Figura 4-6 – Comparativo do sensor de tensão x multímetro	64
Figura 4-7 – Osciloscópio utilizado para verificar o sinal de saída do transformador	65
Figura 4-8 – Resultado da coleta da corrente do ferro elétrico e tensão elétrica da rede.....	66
Figura 4-9 – Protótipo coletando e enviando dados para a página web	67
Figura 4-10 – Página HTML com os dados coletados.....	68

LISTA DE CÓDIGOS

Código 3.1 – Detectando o resultado da temperatura	47
Código 3.2 – Detectando o resultado da corrente.....	48
Código 3.3 – Cálculo do consumo de energia elétrica diário.....	49
Código 3.4 – Página HTML.....	51
Código 3.5 – Configuração da rede local e inicialização do servidor web	52
Código 3.6 – Declaração da EmonLib.h.....	55

LISTA DE EQUAÇÕES

Equação 1 – Conversão para graus Celsius	28
Equação 2 - Cálculo da potência elétrica	39

LISTA DE SIGLAS E ABREVIATURAS

TI – Tecnologia da Informação

V – Volts

GND – Ground

A – Ampère

PBE – Programa Brasileiro de Etiquetagem

ENCE – Etiqueta Nacional de Conservação de Energia

IP – Internet Protocol

HTML – HyperText Markup Language

SPI - Serial Peripheral Interface

RESUMO

Com o avanço da tecnologia muitas empresas do ramo da Tecnologia da Informação utilizam diversos equipamentos robustos e modernos para suprir a necessidade de demandas e armazenamento cada vez maiores. A necessidade de informação a todo instante exige das empresas de Tecnologia a aquisição destes equipamentos, além da manutenção e cuidados com uma boa relação custo-benefício. Este projeto propõe o desenvolvimento de um sistema de monitoramento do consumo de energia e temperatura de racks de rede em tempo real. O objetivo é tornar possível e eficiente o fornecimento de informações confiáveis sobre os equipamentos que estão sendo utilizados em uma empresa. O sistema permite que sejam tomadas ações antecipadas de alerta sobre os equipamentos, principalmente equipamentos que são usados frequentemente e que precisam de manutenção, além de promover a sustentabilidade dentro de empresas. A solução é implementada com a utilização da tecnologia Ethernet, aliada a um microcontrolador ATmega328, com sensores de temperatura ambiente e medição de corrente elétrica.

Palavras chaves: Tecnologia da Informação, Racks, Monitoramento, Ethernet, Sustentabilidade, Microcontrolador.

ABSTRACT

With the development of technology, many companies in the Information Technology industry use modern equipment to meet the need of demands and increasing storage. The need for information every minute requires companies to purchase this equipment, in addition to the maintenance and care of it with a good cost-benefit. This project proposes the development of a monitoring system of energy consumption and temperature of network racks in real time. The goal is to make it possible and efficient to provide reliable information on the equipment being used within a company. The system allows taking an early action on the equipment, particularly most used equipment needing maintenance, and promotes sustainability in companies. The solution is implemented using Ethernet technology, combined with an ATmega328 microcontroller, with temperature sensor and electric current sensor.

Keywords: Information Technology, Racks, Monitoring, Ethernet, Sustainability, Microcontroller.

CAPÍTULO 1 INTRODUÇÃO

Nos últimos 30 anos, os impactos decorrentes da Tecnologia da Informação (TI) atingiram diretamente quase todos os setores da economia mundial. O termo é referente ao conjunto de todas as atividades e soluções que são providas por recursos computacionais, tanto de hardware (computadores, placas, monitores) quanto de software (banco de dados, ferramentas e aplicações). (Rezende & Abreu, 2013)

Com o avanço da tecnologia e da necessidade de uma busca por novas tecnologias inovadoras no mercado mundial, empresas de pequeno e grande porte passaram a não só investir em desenvolvimento de softwares inteligentes e capazes de solucionar problemas complexos, como também em super equipamentos capazes de suportar estes softwares robustos. O grande volume de dados presente nos dias atuais, tanto em empresas como em eletrônicos pessoais, exige um hardware compatível e robusto.

A proposta do trabalho é contribuir para a sustentabilidade por meio da Tecnologia da Informação Verde (TI Verde). Nos dias atuais, a preservação do meio ambiente e a sua sustentabilidade ganham força em diversos ramos da economia mundial. O principal foco da TI Verde está relacionado ao consumo eficiente de energia, além da vida útil dos equipamentos. (Makower, 2009). A TI Verde também sugere a identificação de servidores que produzam mais com menos energia para contribuir com a diminuição dos impactos ao meio ambiente. (Jones, 2010)

As práticas relacionadas à Tecnologia da Informação Verde se tornam cada vez mais necessárias e adequadas para o desenvolvimento da sociedade. A emissão dos gases poluentes que são responsáveis pelo efeito estufa são de responsabilidade de diversas empresas, e estas tem o dever de recompensar e diminuir este impacto ao meio ambiente de forma direta ou indireta. Por isso, o conceito de TI Verde vem sendo algo debatido e aplicado por inúmeras áreas dentro da Tecnologia da Informação. (Richter, 2012)

O projeto tem como tema de estudo a construção de um sistema de monitoramento automatizado de consumo de energia e temperatura. O estudo envolve a integração do conhecimento de tecnologia da informação e engenharia de computação, focada em uma área específica conhecida como automação, com foco em empresas de tecnologia da informação e telecomunicações que utilizam hardware robustos de baixo e alto custo.

A maior vantagem da plataforma Arduino sobre outras plataformas é o seu hardware já ser modelado e totalmente reconfigurável sem o conhecimento necessário de eletrônica digital e arquitetura de microcontroladores. Outra vantagem do Arduino é facilidade de produção e elaboração de softwares por ser uma plataforma Open Source (código fonte livre e aberto). Por conta disso, existe uma grande massa de desenvolvedores dispostos a auxiliar outras pessoas na programação de novos programas para o Arduino. (McRoberts, 2011)

A rede sem fio é utilizada em grande escala e vem sendo aprimorada no passar dos últimos anos. A tecnologia wireless (sem fio) é usada em diversos dispositivos e o seu funcionamento ocorre basicamente através da transformação dos dados de rede em ondas de rádio e em seguida essas ondas são transmitidas por meio de antenas. Cada vez mais este tipo de tecnologia é introduzido no mercado por ser uma tecnologia de fácil uso e instalação. (Torres, 2014)

O sistema de monitoramento remoto do consumo de energia e temperatura em racks de datacenter é capaz de coletar e enviar os dados para uma estação de monitoramento através da tecnologia wireless. Os dados coletados pelos sensores são enviados para a estação de monitoramento através de um módulo Ethernet acoplado à plataforma Arduino e um roteador de rede.

1.1 Objetivos do trabalho

Objetivo Geral – Desenvolver e implementar uma solução que forneça dados de temperatura, corrente e tensão que auxiliem empresas no monitoramento do consumo de energia e temperatura de racks em datacenter;

Objetivos específicos:

- Desenvolvimento do software para monitoramento de equipamentos robustos;
- Coletar dados dos equipamentos através dos sensores;
- Enviar dados coletados através da tecnologia wireless;
- Receber os dados coletados em uma estação de monitoramento via wi-fi;
- Monitorar os dados recebidos por meio de uma interface web;
- Desenvolver interface web utilizando a linguagem de programação HTML;
- Realizar testes do protótipo a fim de garantir a coleta dos dados necessários para o sistema de forma eficaz.

1.2 Motivação

Este projeto tem como motivação a construção de um sistema capaz de monitorar diversos equipamentos de Tecnologia da Informação e seus respectivos locais de instalação, com objetivo de auxiliar na previsão de acidentes, tornando possível antecipar medidas de segurança. Além de contribuir para o meio ambiente e sustentabilidade através do conceito de Tecnologia da Informação Verde.

O intuito é, além da elaboração da solução, construir um protótipo eficiente, pois o objetivo inicial é tornar o sistema acessível para empresas de pequeno e grande porte que buscam, através da sustentabilidade, auxiliar o meio ambiente.

1.3 Metodologia

O projeto abrange a construção de um sistema automatizado com transmissão de informações de maneira remota, com o uso da tecnologia Ethernet. O projeto consiste em desenvolver duas estações, uma de coleta de dados, onde é composta por sensores que coletam as informações, e outra de monitoramento, via interface web, destes dados que são enviados através de um roteador.

A primeira etapa consiste em estudar e coletar todas as referências bibliográficas usadas para o desenvolvimento do projeto, sendo elas, sobre Arduino Uno, módulo Ethernet para Arduino Uno, equipamentos de Tecnologia da Informação, Tecnologia da Informação Verde, linguagem de programação Wiring (linguagem baseada em C/C++), sensor de corrente contínua, sensor de temperatura e sensor de tensão;

A segunda etapa propõe em desenvolver uma placa de circuito impresso com os sensores controlados pelo Arduino Uno. Nessa etapa serão realizadas todas as ligações dos circuitos eletrônicos e sensores com o micro controlador Arduino Uno, assim como o módulo Ethernet que será acoplado à plataforma Arduino. Essa etapa também consiste em desenvolver a página web em linguagem HTML que disponibiliza os dados coletados para a estação de monitoramento em rede local;

A terceira etapa consiste em desenvolver o software para o sistema automatizado, que é utilizado para monitoramento dos equipamentos de Tecnologia da Informação. O software, juntamente com o hardware, é capaz de

fazer a leitura dos dados coletados através dos sensores instalados na placa de prototipagem, processar as informações no microcontrolador e enviar via módulo Ethernet para um roteador distribuir essas informações para a estação de monitoramento;

A quarta etapa consiste em realizar a integração do software desenvolvido com o hardware, onde o código desenvolvido é transferido para o microcontrolador Arduino Uno. Esta fase também consiste em efetuar os ajustes que se julgam necessários;

A quinta e última fase consiste em realizar os testes necessários do sistema de monitoramento remoto para verificar a integridade de todo o sistema desenvolvido, desde testes de ligação do circuito até testes dos códigos desenvolvidos. Nesta fase, os testes realizados consistem em concretizar o funcionamento dos sensores e microcontrolador, utilizando equipamentos com diferentes amperagens em ambientes de diversas temperaturas.

1.4 Resultados Esperados

Espera-se com este projeto construir um sistema de monitoramento remoto para equipamentos de Tecnologia da Informação, por meio da elaboração de um sistema baseado na plataforma Arduino Uno, que consiga coletar os dados de forma rápida, que não ultrapasse 7 segundos no tempo de resposta dos sensores, e de forma precisa, que não ultrapasse uma margem de erro de 2%, através dos sensores nele instalados, e enviá-los via roteador de rede local, para uma interface web desenvolvida em HTML.

Construir um sistema de fácil instalação para que seja acessível para diversas empresas, afim de evitar acidentes e perda de equipamentos, e também promover a sustentabilidade, mapeando e identificando servidores que consomem energia em excesso, acima de aproximadamente 200 kW/dia, afim de contribuir com o meio ambiente e a Tecnologia da Informação Verde.

Atingir resultados positivos através dos testes realizados para comprovar o correto funcionamento do sistema de monitoramento remoto de consumo de energia e temperatura com uma margem de erro de no máximo 3% na coleta das informações.

1.5 Estrutura do trabalho

Segue descrição para o restante deste trabalho:

O Capítulo 2 apresenta o referencial teórico dos sensores, Arduino, módulo Ethernet e linguagem de programação.

O Capítulo 3 detalha a solução proposta, mostrando o desenvolvimento do projeto e ilustrando o modelo utilizado;

O Capítulo 4 descreve a análise dos resultados obtidos e aplicação do sistema proposto, detalhando todos os testes realizados durante a construção do protótipo do projeto;

O Capítulo 5 tem por objetivo apresentar as conclusões finais obtidas após o término do desenvolvimento do projeto;

CAPÍTULO 2 REFERENCIAL TEÓRICO

Este capítulo apresenta os resultados de todas as pesquisas feitas que serviram como bases teóricas para o desenvolvimento do sistema de monitoramento remoto proposto.

2.1 Tecnologia da Informação Verde

A Tecnologia da Informação Verde é uma tendência mundial que é orientada aos impactos dos recursos tecnológicos no meio ambiente. A preocupação deste conceito está relacionada desde o consumo consciente de energia por servidores e redução de matéria-prima (toners e papéis) até o descarte de equipamentos por meio da reciclagem e o reuso de máquinas. A tendência da TI Verde também cresce de forma exponencial, pois a preocupação ambiental é discutida em vários ramos da economia mundial. Com essa expansão da Tecnologia da Informação Verde, empresas de baixo e grande porte passaram a não só alcançar ganhos ambientais, como também ganhos econômicos ao optar por medidas que buscam a sustentabilidade. (Jones, 2010)

Neste sentido, o mundo corporativo desperta para exercer, e principalmente, elaborar certas ações para satisfazer as exigências de um negócio sustentável. Por isso, a importância para a sociedade, negócios e futuro do meio ambiente efetiva de vez a prática da Tecnologia da Informação Verde com o objetivo de ganhar mais espaço e interesse entre os profissionais de TI para o desenvolvimento de inovações tecnológicas que facilitem a sustentabilidade. (Makower, 2009)

A proposta do projeto é desenvolver um sistema de monitoramento remoto capaz de contribuir para a sustentabilidade do meio ambiente através do conceito de Tecnologia da Informação Verde, para que seja possível alertar empresas de tecnologia quais equipamentos consomem energia em excesso.

Como a Tecnologia da Informação Verde está sendo o foco de muitas empresas que desejam diminuir o impacto de seus trabalhos e recursos tecnológicos ao meio ambiente, a ideia do projeto é ajudar e contribuir ainda mais para estas empresas que estão aderindo à essa tendência global.

2.2 Consumo de Energia Elétrica

O consumo de energia elétrica aumenta exponencialmente em todo o mundo devido à aceleração do consumismo em escala global. Por isso, novas hidrelétricas são construídas para suprir a alta demanda de energia. As hidrelétricas, apesar de não poluírem o ar, geram impacto diretamente ao meio ambiente. O impacto no meio ambiente referente ao alto consumo de energia começa a ganhar força a partir do século XX. (Westleya, 2009)

Efeito estufa e o desmatamento são alguns dos exemplos que aumentaram nos últimos 100 anos devido ao alto consumo de energia. Medidas vem sendo tomadas com o objetivo de diminuir esse alto consumo de energia. No Brasil foi desenvolvido um projeto de etiquetagem (PBE – Programa Brasileiro de Etiquetagem) que indica o nível de consumo de energia de cada equipamento eletrônico. A ENCE (Etiqueta Nacional de Conservação de Energia) classifica os eletrônicos em categorias de “A” (mais eficiente) até “E” (menos eficiente), além de proporcionar outras informações relevantes. (INMETRO/PBE, 2015)

2.3 Racks de DataCenter

DataCenters possuem um papel importante no funcionamento de diversos serviços da nossa rotina. DataCenter é um ambiente específico onde são concentrados servidores, equipamentos de processamento e armazenamento de dados, switches de redes, roteadores, entre outros. Com o objetivo de armazenar milhões de servidores de banco de dados, estes equipamentos geralmente são montados em racks ou armários metálicos.

Estes ambientes são responsáveis por diversas áreas da economia global, como por exemplo telecomunicações, internet, energia, sistemas de segurança, entre outros. Por isso a importância de monitorar estes equipamentos, assim como a temperatura em que os mesmo estão instalados para não ocorrer qualquer tipo de dano.

2.4 Descrição dos Materiais Utilizados

2.4.1 Plataforma Arduino

Neste projeto, o principal componente do circuito é a placa de microcontrolador da plataforma Arduino Uno, que utiliza o microcontrolador Atmega328. Uma placa é utilizada para controlar os sensores e processar as informações recebidas. O Arduino Uno possui conexão através de uma porta serial USB, 14 entradas e saídas digitais, 6 entradas analógicas e um conector DC Jack de alimentação. (ARDUINO.CC)

A descrição detalhada do hardware da plataforma Arduino UNO segue abaixo:

- Microcontrolador: ATmega328;
- Tensão de operação: 5V;
- Tensão de entrada (recomendada): 7-12V;
- Tensão de entrada (limites): 6-20V;
- Pinos de entrada e saída digital: 14;
- Pinos de entrada analógico: 6;
- Corrente contínua por pino entrada/saída: 40 mA;
- Corrente contínua para o pino de 3.3V: 50 mA;
- Memória *Flash*: 32 KB (ATmega328);
- SRAM: 2 KB (ATmega328);
- EEPROM: 1 KB (ATmega328);
- Velocidade do *Clock*: 16 MHz.

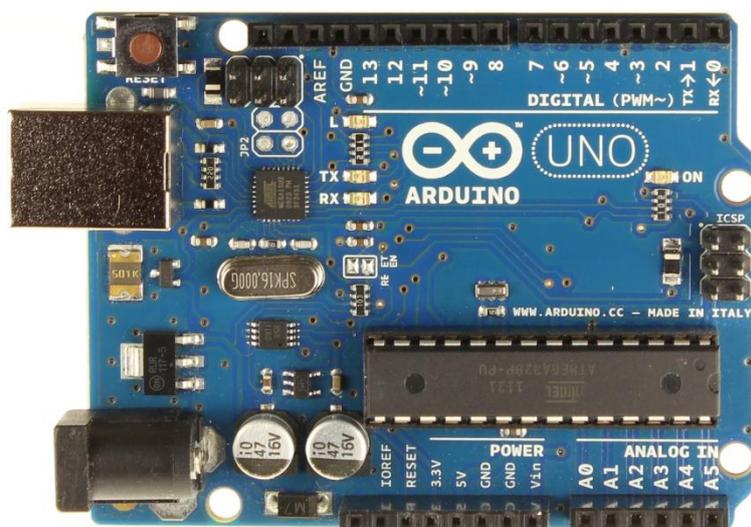


Figura 2-1 – Arduino UNO

(Fonte: (ARDUINO.CC))

O Arduino Uno pode receber uma alimentação que varia de 7 a 12 volts devido à um regulador de tensão de 5 volts que é capaz de converter a tensão de entrada para uma tensão constante de 5V. A tensão de entrada utilizada no projeto é de 9V (dentro dos limites recomendados 7-12V). A placa microcontroladora Arduino Uno apresenta diversas características positivas, além de fácil manuseio, é *open source*, o que facilita a customização da placa caso seja necessário. (ARDUINO.CC) A figura 2-1 mostra a plataforma Arduino UNO.

O microcontrolador ATmega328 tem um conversor A/D (analógico/digital) de 10 bits de resolução para os pinos de entrada analógica, o que representa 1024 valores diferentes criados para um sinal analógico de entrada. (ARDUINO.CC)

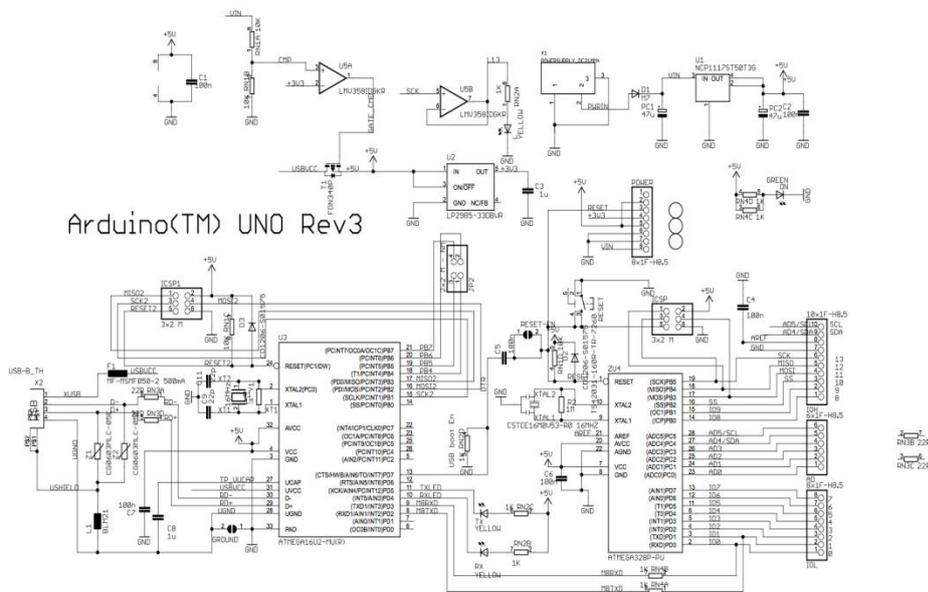


Figura 2-2 – Esquema elétrico da plataforma Arduino UNO

(Fonte: (ARDUINO.CC))

A figura 2-2 mostra o esquema elétrico da plataforma Arduino UNO. O esquema elétrico é dividido em três partes: alimentação, processador USB e processador principal. O processador USB é o responsável pela comunicação entre a plataforma e o computador utilizado para programação do código, o microcontrolador Atmega328 não suporta conexão direta com a porta USB, por isso este módulo é necessário para realizar esta interligação. A principal função deste módulo é converter os dados do USB do computador para um sinal serial e após essa conversão, os dados poderão ser lidos pelo processador principal da plataforma Arduino. Por isso, o funcionamento do processador USB opera como um conversor USB-Serial. O processador utilizado para exercer esta operação é o Atmega16U2.

Já o processador principal é o “cérebro” da plataforma, onde é processada toda a informação principal, como por exemplo, leitura de sensores e integração com *shields* (elementos externos). Além de executar todo o software que está programado nele. E finalmente, este processador é responsável por receber, enviar e interpretar os sinais que são recebidos pelo processador USB Atmega16U2. O processador utilizado na plataforma Arduino UNO para esta função é o processador Atmega328.

O processador principal também contém um oscilador de frequência 16 MHz, o conector de programação que é utilizado para uma programação de um software prévio da plataforma Arduino ainda na própria fábrica do Arduino, este software é conhecido como o *bootloader* e é responsável por simplificar os softwares para o chip de memória flash que está embutido na plataforma Arduino. Ainda neste módulo, existe um botão *reset* para reiniciar o processador principal e por fim o circuito dos conectores para shields externos, como por exemplo o shield Ethernet que é usado neste projeto. Os conectores são ligados diretamente aos pinos de entrada e saída da plataforma Arduino assim como as tensões de alimentação e aterramento.

Finalmente, o módulo da alimentação, que contém um regulador de tensão, que é responsável por estabilizar a tensão que pode variar entre 7 e 20 volts para 5V, que é a tensão necessária para o funcionamento da plataforma Arduino. Este módulo também contém um LED para indicar quando a plataforma está ligada. (ARDUINO.CC)

2.4.2 IDE do Arduino

O software do Arduino, o Arduino IDE (Integrated Development Environment), é de fácil uso e através dele é possível programar o código e fazer upload para a placa. O Arduino IDE ainda pode ser utilizado com qualquer tipo da plataforma Arduino. (McRoberts, 2011)

O software também disponibiliza bibliotecas prontas que facilitam a programação, mas também é possível criar bibliotecas pessoais e adicioná-las no software. A linguagem de programação utilizada para desenvolver os códigos é baseada em C e C++, porém esta linguagem possui o nome de Wiring. (Monk, 2013) A figura 2-3 mostra a interface do IDE.

Para o desenvolvimento do projeto, a biblioteca para utilização do shield Ethernet é utilizada para facilitar a comunicação da plataforma Arduino

com este componente externo assim como a biblioteca EmonLib.h que já contém os algoritmos necessários para medição da corrente e tensão elétrica.

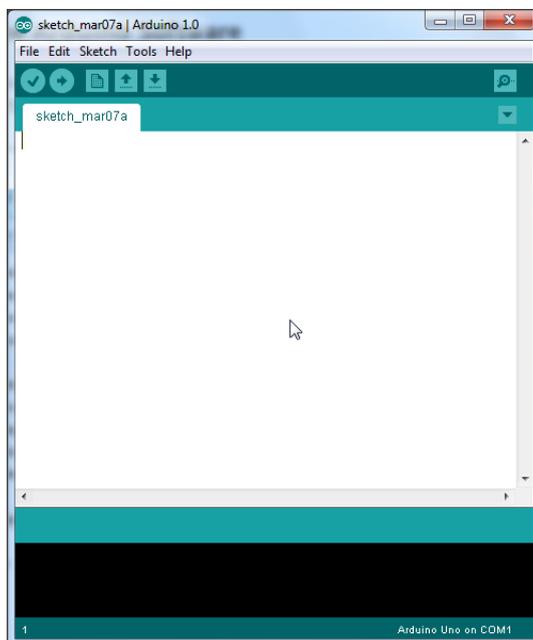


Figura 2-3 – IDE do Arduino
(fonte: Autor)

A Figura 2-3 mostra o software utilizado para desenvolver o código do sistema de monitoramento remoto e enviá-lo para a plataforma Arduino poder processar as informações requisitadas. Nesta interface, ainda é possível testar as informações que estão sendo coletadas por sensores através de uma interface visual oferecida por este software.

2.4.1 Sensor de temperatura

Para fornecer a temperatura do ambiente onde os equipamentos de Tecnologia da Informação estão alocados, é necessário a utilização de um sensor de temperatura, conhecido como LM35. O sensor LM35 é fabricado pela National Semiconductor, e é um sensor de alta precisão. O LM35 não necessita de nenhum tipo de calibração antes de ser utilizado, porém, para ser

lido pelo Arduino, é necessário uma pequena conversão visto que o Arduino lê dados inteiros entre 0 e 1023. Este sensor possui uma resolução de 10mV para cada grau Celsius, e como é alimentado por um Arduino que fornece 5V, é possível chegar na equação 1 para encontrar o valor exato da temperatura que está sendo verificada. (TEXAS INSTRUMENTS, 2013)

$$temperatura = (valor\ lido * 500 / 1023)$$

Equação 1 – Conversão para graus Celsius

O sensor LM35 apresenta uma precisão de temperatura que pode variar de $\frac{1}{4}$ °C a $\frac{3}{4}$ °C, dentro de uma faixa de temperatura de -55°C à 150°C, o que atende à solução proposta. Para este projeto o sensor de temperatura utilizado é o de encapsulamento TO-92 por ser o mais comum no mercado, porém é possível encontrar outros tipos de encapsulamento para o sensor LM35 como por exemplo o TO-46 que apresenta um tempo de absorção da temperatura maior do que o TO-92 conforme mostrado no gráfico abaixo.

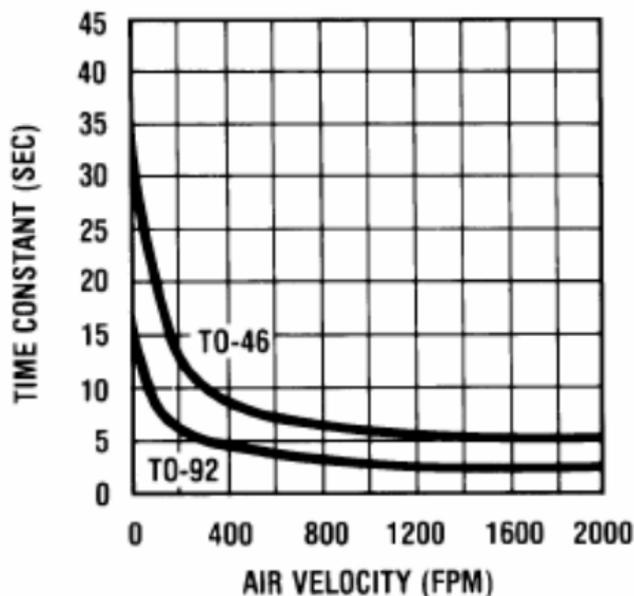


Figura 2-4 – Gráfico de constante térmica do tempo

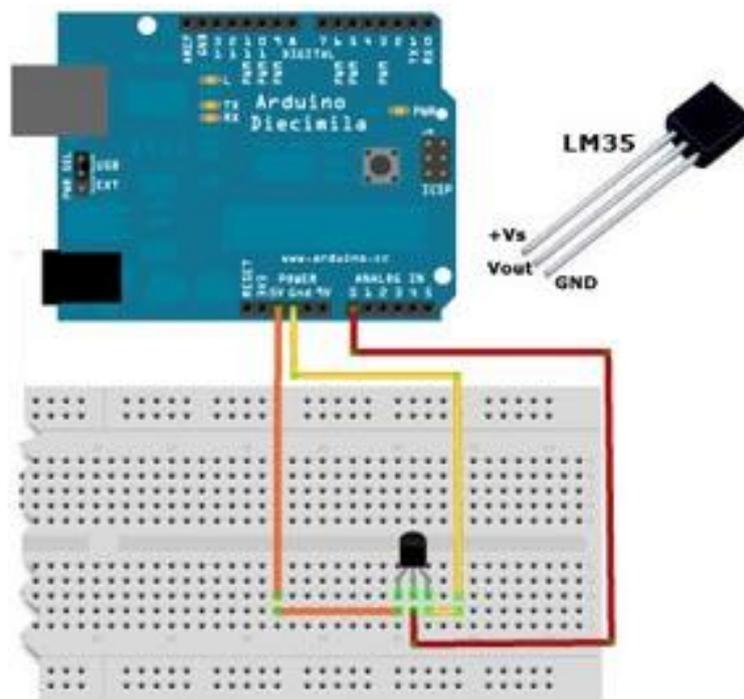
(Fonte: (National Semiconductor))

A figura 2-4 mostra o gráfico de constante térmica do tempo do sensor de temperatura LM35. Este gráfico mostra o tempo necessário para que o componente absorva todo o calor do ambiente pra que, em seguida possa fornecer uma informação precisa de medição.

Para o TO-92, este tempo é de aproximadamente 16 segundos. Ou seja, o sensor de temperatura utilizado neste projeto irá levar cerca de 16 segundos para absorver qualquer alteração térmica do ambiente para em seguida enviar a tensão de saída padrão do TO-92 de 10mV por grau Celsius. E finalmente esta tensão de saída ser conectada em uma porta analógica do microcontrolador para ser lido e convertido para grau Celsius conforme mencionado no início desta seção. O tempo necessário para o sensor de temperatura realmente fornecer informações precisas caso a temperatura do ambiente seja alterada, não interfere de forma significativa neste sistema.

A figura 2-5 detalha como são feitas as ligações do Arduino com o sensor de temperatura LM35.

- O pino positivo do sensor que será ligado em 5V;
- O pino de dados analógicos do sensor que será ligado à um pino analógico do Arduino;
- O pino GND do sensor que será ligado no GND do Arduino;



**Figura 2-5 – Conexões do sensor LM35 com Arduino
(fonte: (ladyada.net, 2012))**

Características principais do sensor LM35:

- É calibrado diretamente em graus Celsius (centígrados);
- Fator de escala linear de + 10,0 mV / °C;
- 0,5 °C de precisão para temperaturas superiores a 25 °C;
- Opera com a faixa de temperatura de -55 ° a 150 ° C;
- Opera de 4 a 20 Volts;
- Necessita de uma corrente de 60mA;
- Baixo auto-aquecimento, 0,1 °C (no ar);
- Impedância de saída baixa, de 0,1 Ω para 1mA de carga.

2.4.2 Sensor de corrente

O fluxo ordenado de cargas elétricas é denominado de corrente elétrica. A corrente elétrica é definida como a razão entre a quantidade de

carga que atravessa certa seção transversal do condutor em um certo intervalo de tempo. A unidade de medida para representar a corrente elétrica é o Coulomb por segundo (C/s), mais conhecido como Ampère (A) no Sistema Internacional de Unidades em homenagem ao físico e matemático francês André-Marie Ampère (1775-1836). (Cruz, 2014)

No protótipo deste projeto é utilizado um sensor de corrente ACS712-30A que é fabricado pela Allegro MicroSystems LLC, uma empresa americana líder em desenvolvimento e produção de semicondutores de alta performance.

O sensor ACS712 fornece informação precisa de uma corrente contínua ou alternada, além de ser de fácil implementação. O ACS712 também possui uma resistência interna de $1.2\text{m}\Omega$, que proporciona uma baixa perda de potência.

A figura 2-6 mostra como o sensor de corrente ACS712 é conectado à plataforma Arduino. (Allegro MicroSystems LLC, 2013)

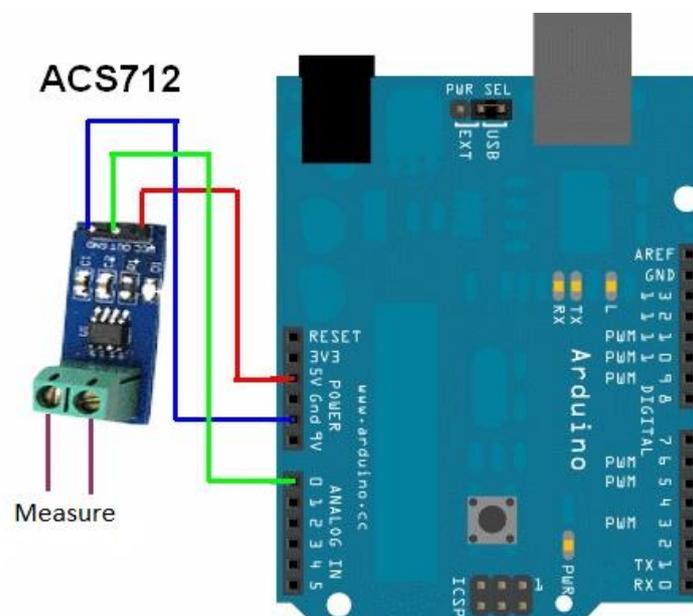


Figura 2-6 – Conexões do sensor ACS712 com Arduino
(fonte: (helpdev.com.br, 2014))

O sensor de corrente possui um circuito integrado de 8 pinos conforme mostra a figura 2-7. Os pinos 1, 2, 3 e 4 são os terminais onde é colhido as amostras de corrente, o pino 5 representa o aterramento do circuito e o pino 6 é utilizado para definição da largura de banda através de um capacitor externo. Já o pino 7 é o pino da saída analógica e o pino 8 o de alimentação do circuito integrado. (Allegro MicroSystems LLC, 2013)

Este circuito integrado opera de forma automática dentro do sensor de corrente sem se fazer necessário qualquer tipo de ajuste para o seu funcionamento.

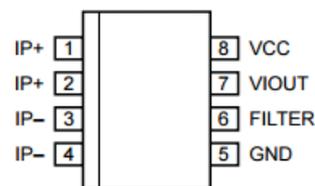


Figura 2-7 – Diagrama do circuito integrado
(fonte: (alldatasheet.com, 2014))

A figura 2-8 mostra o diagrama de blocos detalhado do circuito integrado que compõe o sensor de corrente ACS712-30A, detalhando todas as resistências do circuito integrado, além dos amplificadores, capacitores e os pinos de entrada até a chegada do sinal lido que é efetuada no pino 7, que também é o pino de saída analógico. Este circuito, conforme mencionado, não necessita de nenhum tipo de ajuste antes do uso do sensor de corrente por ser um circuito integrado já presente no sensor. Este circuito integrado é responsável por receber toda a corrente do equipamento que está sendo monitorado e em seguida enviar o sinal de saída para a plataforma Arduino. O circuito necessita de uma alimentação que varia entre 4.5V e 5.5V e um aterramento para o funcionamento correto. No caso deste projeto, alimentação utilizada é de 5V, por ser a alimentação promovida pela plataforma Arduino.

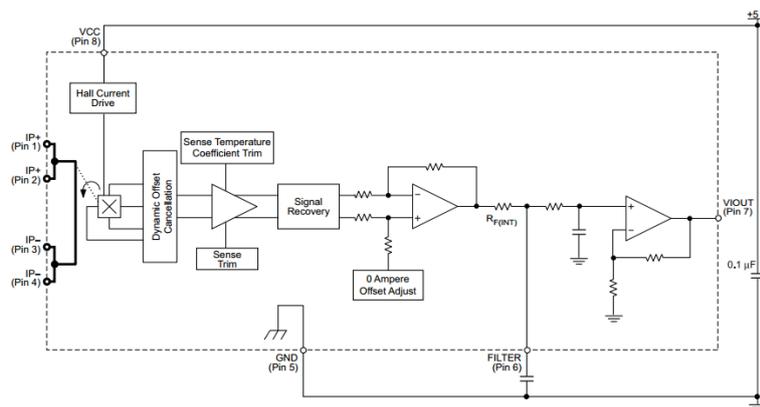


Figura 2-8 – Diagrama do circuito integrado
(fonte: (alldatasheet.com, 2014))

A figura 2-9 mostra o gráfico de performance para a corrente de saída versus a temperatura do ambiente. O sensor de corrente não sofre grandes variações entre temperaturas de 25°C e 50°C, o que não influencia na aplicação do sistema de monitoramento remoto proposto. Apenas temperaturas extremas, conforme o gráfico, alteram o correto funcionamento do sensor de corrente. Porém, a performance do sensor não chega a ser tão prejudicada em temperaturas baixas ou elevadas, de acordo com a figura percebe-se que a corrente de saída irá variar em menos de 1 A para casos extremos.

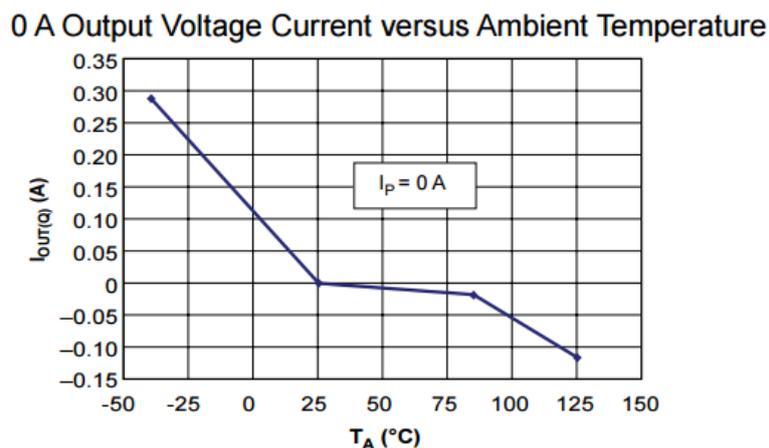


Figura 2-9 – Diagrama do circuito integrado
(fonte: (alldatasheet.com, 2014))

Algumas das principais características do ACS712-30A seguem abaixo:

- Faixa de leitura da corrente bidirecional: -30A a +30A;
- 1.2m Ω de resistência do condutor interno;
- Alimentação de operação de 4.5V – 5.5V;
- 66-185mV de sensibilidade de saída;
- Tempo de resposta de saída de 5 μ s após uma entrada de corrente;
- Tensão de saída: Saída Analógica de 66mV/A;
- Tensão de isolamento: 2.1 KVRMS;
- Largura de banda: 80kHz.

2.4.3 Ethernet Shield

Para enviar os dados coletados para a estação de monitoramento do sistema remoto, é necessário a utilização do Ethernet Shield. Este módulo consiste em tornar possível uma conexão do Arduino com um roteador e transmitir os dados via cabo Ethernet. Com a instalação e configuração deste módulo, também é possível ler as informações dos sensores de qualquer ponto da rede através de uma interface web desenvolvida em HTML. Essa placa tem a vantagem de ser de fácil integração com a plataforma Arduino Uno, pois tem o formato de um shield (formato empilhável), onde é encaixada sobre a plataforma Arduino, conforme é ilustrado na figura 2-10.

O módulo Ethernet Shield baseia-se no chip Wiznet W5100, que fornece uma rede capaz de trabalhar com protocolos de rede da camada de transporte (TCP e UDP). Assim como a plataforma Arduino, o Ethernet Shield possui documentação acessível para todos e uma biblioteca exclusiva para desenvolvimento de códigos de acordo com a necessidade de cada projeto. O módulo apresenta uma entrada de conexão padrão de rede (conector RJ-45)

que torna possível o uso de cabos de rede comuns no mercado. (ARDUINO.CC)

Para o desenvolvimento da solução proposta, o Ethernet Shield é utilizado para transmitir os dados coletados dos sensores e enviar-los através de um cabo de rede RJ-45 para um roteador, para que este possa fornecer em rede local as informações necessárias para monitoramento das informações em uma interface Web. Para a instalação deste shield à plataforma Arduino UNO, basta realizar o encaixe entre os pinos de entrada, conforme mencionado no referencial teórico deste trabalho.



**Figura 2-10 – Conexão do Ethernet Shield com o Arduino
(fonte: (waihung.net))**

Seguem as características detalhadas do Ethernet Shield:

- Conector RJ45;
- Soquete para cartão micro-SD;
- Controlador Ethernet W5100 com buffer interno de 16k;
- Velocidade de conexão de 10/100Mb;

- Tensão de operação de 3,3 – 5V;
- Suporta até 8 conexões TCP/UDP simultâneas;
- Compatível com a biblioteca Ethernet do Arduino;
- Compatível com Arduino Uno (usado no projeto);

CAPÍTULO 3 DESENVOLVIMENTO DO PROTÓTIPO

Este capítulo tem como objetivo mostrar todo o desenvolvimento necessário para a construção do protótipo do sistema proposto, tendo como base os referenciais teóricos apresentados no capítulo 2.

3.1 Apresentação Geral do Sistema Proposto

As etapas a seguir mostram o desenvolvimento da estação de coleta de informações para que seja possível contribuir com a sustentabilidade através do conceito de Tecnologia da Informação Verde, assim como também o desenvolvimento da página HTML que é utilizada no sistema de monitoramento remoto para monitorar os dados recebidos na estação de monitoramento.

A figura 3-1 mostra as etapas seguidas para a realização do projeto. A primeira etapa consiste na pesquisa de todas as referências que são utilizadas no projeto proposto. Em seguida, todo o material necessário é adquirido para desenvolvimento da parte do hardware do protótipo. Na terceira etapa, o código do sistema é desenvolvido para finalmente ser integrado e testado na última etapa de testes e resultados.

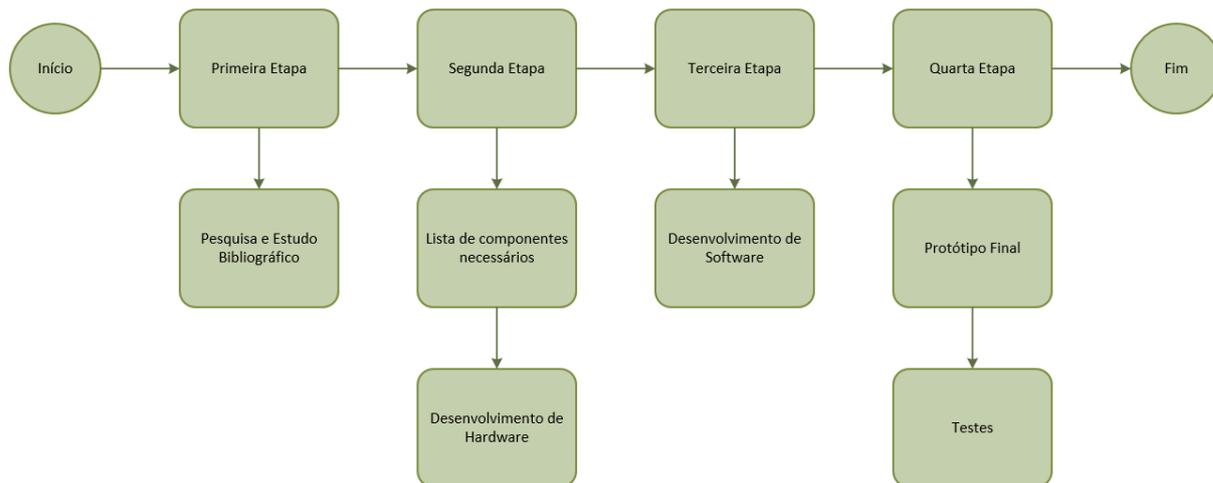


Figura 3-1 – Fluxograma do Projeto
(Fonte: Autor)

Na figura 3-2 é mostrado o diagrama do projeto, desde a coleta dos dados até o monitoramento. Também é apresentado como é feita a transmissão destes dados entre a estação de coleta e estação de monitoramento através do módulo Ethernet e um roteador. Seguem as etapas do sistema de monitoramento proposto e em seguida o detalhamento de cada uma delas.

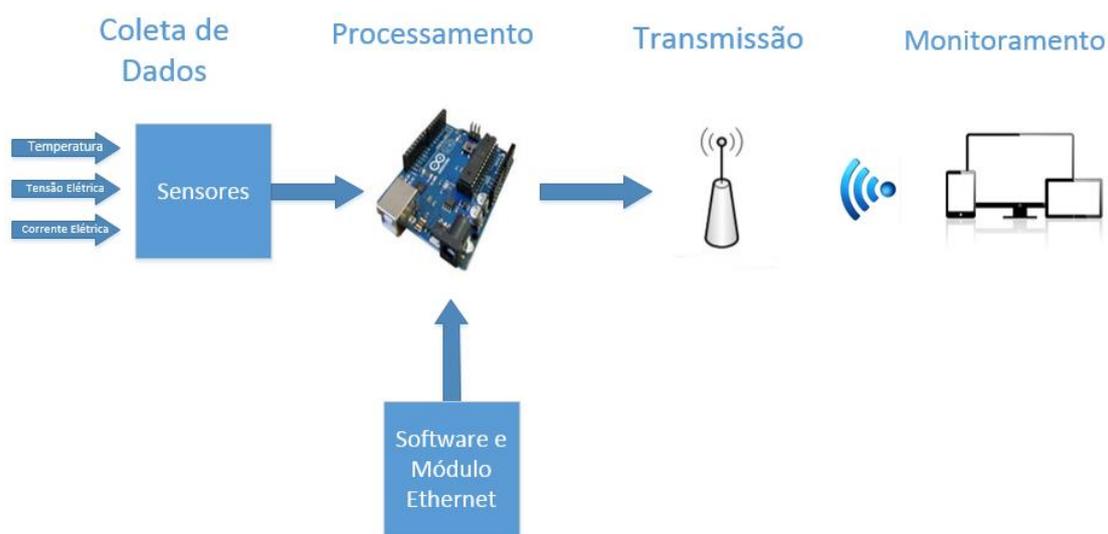


Figura 3-2 – Diagrama do Projeto
(Fonte: Autor)

O primeiro bloco representa a estação de coleta de dados onde é composta pelos sensores responsáveis por coletar as informações do sistema. As informações sobre a temperatura do ambiente e a corrente elétrica do equipamento será coletada neste momento, assim como a tensão elétrica. Em seguida, estas informações são processadas pela plataforma Arduino que contém todo o software de processamento e transmissão destes dados que é desenvolvido utilizando o IDE do Arduino, conforme referenciado no capítulo 2. Além das conversões necessárias para se encontrar o valor correto, como por exemplo, a conversão do sensor de temperatura para graus Celsius conforme referenciado no capítulo 2 deste projeto.

Também nesta etapa, o cálculo para o consumo de energia é realizado pelo microcontrolador, como por exemplo o cálculo da potência elétrica representada pela equação abaixo:

$$P = V \times I$$

Equação 2 - Cálculo da potência elétrica

A potência elétrica irá retornar um valor em Watts, correspondente ao consumo instantâneo de energia do rack, já a temperatura será convertida para graus Celsius conforme mencionado anteriormente.

A plataforma Arduino também conta com um módulo Ethernet, que é necessário para envio dos dados coletados através da tecnologia Ethernet utilizando um cabo simples de rede RJ-45. Em seguida, este cabo é conectado à um roteador, representado pela estação de transmissão no diagrama acima, para que seja possível enviar as informações coletadas e processadas através da tecnologia Wireless em rede local. Para o envio destas informações, o software é configurado com as informações sobre a rede local, como por exemplo, endereço IP, máscara de sub rede, endereço MAC e gateway coletados pelo computador acessando o roteador da rede local.

Finalmente, na estação de monitoramento os dados são recebidos por um dispositivo que possua tecnologia Wireless, os dispositivos se conectam na rede local que é criada pelo roteador e em seguida é possível verificar as informações coletadas pelos sensores, como por exemplo, temperatura, corrente elétrica em Amperes, tensão elétrica em Volts e potência elétrica em Watts.

3.2 Estação de coleta de dados

Na física, a temperatura está relacionada com a energia presente em um sistema termodinâmico. Já a medição de corrente elétrica indica a quantidade do fluxo de elétrons ordenados, ou seja, um deslocamento de partículas portadoras de cargas elétricas. Existem dois tipos de classificação da corrente elétrica, são elas contínua ou alternada. A corrente elétrica é classificada como contínua quando este fluxo de cargas ocorre em um único sentido, como por exemplo em baterias ou pilhas. Já a corrente alternada, que é medida neste projeto, indica uma variação no fluxo dessas partículas portadoras de cargas elétricas.

Já a tensão elétrica é a diferença do potencial elétrico entre dois pontos, a unidade de medida utilizada para representação da tensão é o Volt. A diferença deste potencial é basicamente igual ao trabalho que deve ser realizado por cada unidade de carga contra um campo elétrico para mover uma carga qualquer.

Como mencionado anteriormente, esta estação possui dois sensores responsáveis pela coleta de dados dos equipamentos que serão monitorados além de um circuito para calcular a tensão da rede elétrica. Os sensores de temperatura e corrente elétrica, conforme descritos no capítulo 2, são de fácil instalação e configuração. Já o circuito para realizar a medição da tensão elétrica é desenvolvido e programado para coletar o valor da tensão elétrica de uma rede 220V. Por isso, é utilizado um transformador de tensão para

converter a tensão de entrada, por ser uma alta tensão, e conseqüentemente não ser suportada pela plataforma Arduino. Os demais detalhes deste circuito estão nas próximas sessões deste trabalho.

Nesta etapa, é montado e configurado o circuito com os sensores de temperatura e corrente elétrica. Para a montagem inicial, os dois pinos analógicos da plataforma Arduino são conectados, aos sensores de temperatura e corrente elétrica.

A figura 3-3 mostra o circuito do LM35, o pino do aterramento é conectado à porta de aterramento da plataforma Arduino que serve para liberar a eletricidade que se mantém no componente. Já o outro pino é um pino de alimentação que tem como função prover cargas para o sensor de temperatura, e finalmente o último pino que é ligado à uma porta analógica da plataforma Arduino para leitura da temperatura.

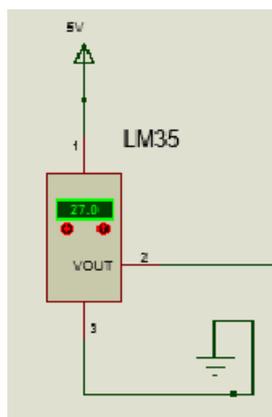


Figura 3-3 – Circuito do sensor LM35
(Fonte: Autor)

Os demais detalhes são descritos no decorrer desta sessão. A figura 3-4 mostra como é feita a montagem detalhada do sensor LM35 à plataforma Arduino. A alimentação do sensor de temperatura é de 5V por ser a alimentação utilizada na saída da plataforma Arduino. O aterramento também é utilizado de um pino da plataforma Arduino.

O LM35 opera em uma faixa entre 4 e 20 Volts conforme mencionado no capítulo 2, e por isso é possível utilizar a alimentação da plataforma Arduino para realização da leitura da temperatura de forma precisa.

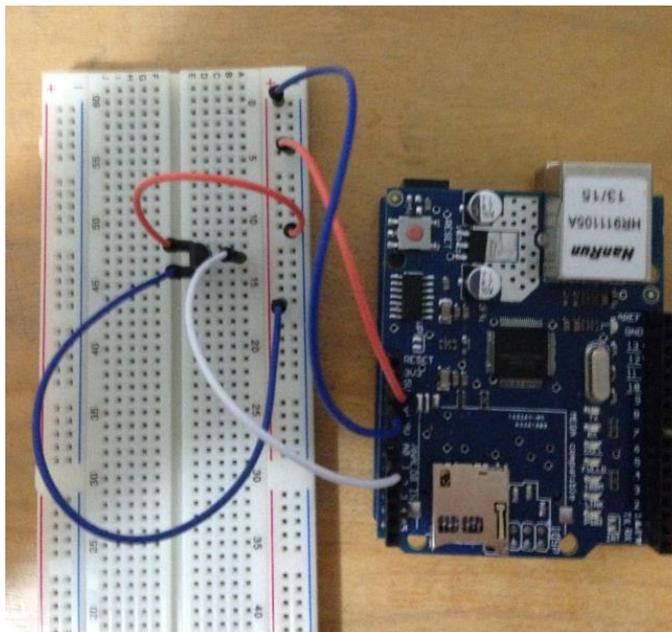


Figura 3-4 – Montagem do sensor LM35
(Fonte: Autor)

O sensor LM35 possui 3 pinos, como referenciado anteriormente no capítulo 2 deste projeto. O pino positivo do sensor é alimentado em 5V provido pela plataforma Arduino (jumper laranja), o pino de dados analógico é vinculado ao pino analógico A0 da plataforma Arduino (jumper branco) e finalmente o pino GND do LM35 é conectado ao GND do Arduino (jumper azul).

Os pinos analógicos da plataforma Arduino são utilizados para ler e processar as informações que são enviadas para esta entrada da plataforma.

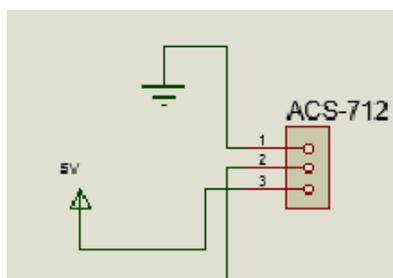


Figura 3-5 – Circuito do sensor ACS712
(Fonte: Autor)

Já a figura 3-5 mostra o circuito elétrico do sensor de corrente elétrica que, assim como o sensor de temperatura, possui três pinos com as

mesmas funções, o pino 1 indica o aterramento do componente, o pino 3 mostra a alimentação do ACS712 e por fim, o pino 2 que indica a saída para leitura da corrente elétrica pela plataforma Arduino.

O sensor de corrente opera apenas a uma tensão entre 4.5 e 5.5V, conforme referenciado no capítulo 2 deste projeto, porém a alimentação fornecida pela plataforma Arduino é de 5V e com isto é possível atender o sensor de corrente.

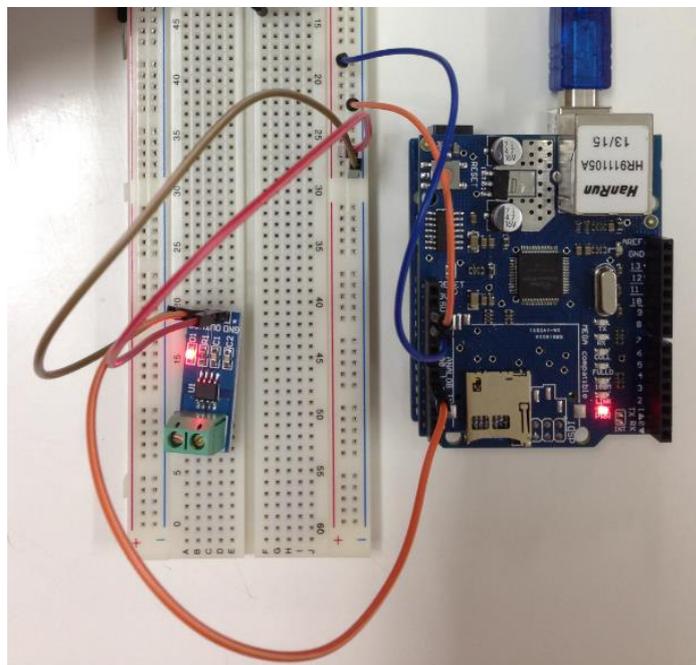


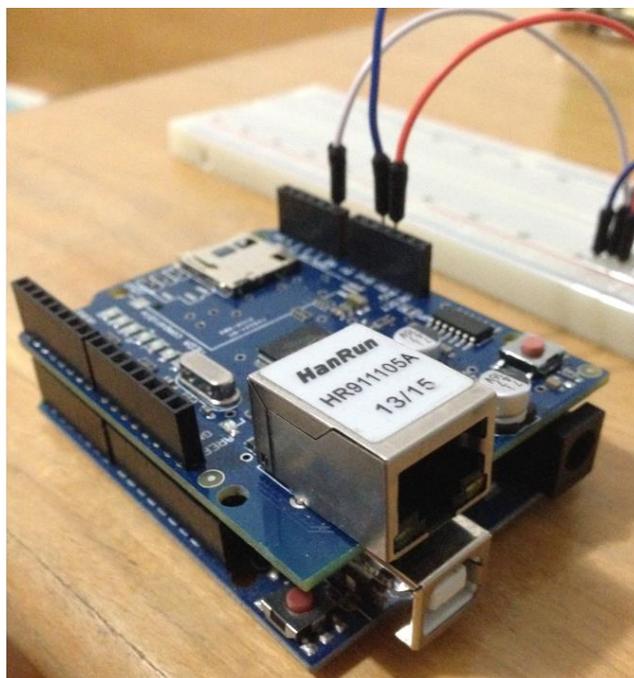
Figura 3-6 – Montagem do sensor ACS712
(Fonte: Autor)

A figura 3-6 mostra como é feita a montagem detalhada do sensor de corrente elétrica junto à plataforma Arduino. O GND do ACS712 é conectado ao GND do Arduino, conforme mostra o jumper vermelho. O *output* do sensor de corrente é ligado ao pino A1 do Arduino para leitura dos dados coletados. Esta ligação é representada pelo jumper laranja. E, finalmente o pino VCC do sensor de corrente é alimentado por uma tensão de 5V do Arduino, conforme é mostrado pelo jumper marrom.

3.3 Módulo Ethernet

Nesta etapa, o módulo Ethernet é acoplado à plataforma Arduino para atender a necessidade do monitoramento dos dados coletados de forma remota. Como mencionado no capítulo 2, o Shield Ethernet torna possível a conexão da plataforma Arduino a um roteador através de um cabo de rede RJ-45. Também nesta fase, todo o software necessário para o processamento das informações registradas pelos sensores é desenvolvido, assim como o código necessário para enviar os dados processados para o roteador através do cabo RJ-45 conectado no Shield Ethernet.

O Shield Ethernet também é de fácil instalação e não necessita de nenhuma configuração a mais para se comunicar com a plataforma Arduino. É necessário apenas a inclusão da biblioteca Ethernet.h, no desenvolvimento do software, para que se torne possível a comunicação, via cabo de rede simples, entre o Arduino e o roteador de rede. O Shield Ethernet também possui um soquete para cartão micro-SD caso seja necessário armazenar alguma informação.



**Figura 3-7 – Montagem do Ethernet Shield ao Arduino
(Fonte: Autor)**

A figura 3-7 mostra o encaixe do Ethernet Shield à plataforma Arduino. Nesta montagem, as portas da plataforma Arduino são as mesmas usadas pelo módulo Ethernet no momento da declaração de variáveis do código, fazendo com que não seja necessária nenhuma configuração a mais para funcionamento dos sensores.

Este encaixe é possível devido a interface SPI (Serial Peripheral Interface), um protocolo responsável pela comunicação entre microcontroladores com outros periféricos.

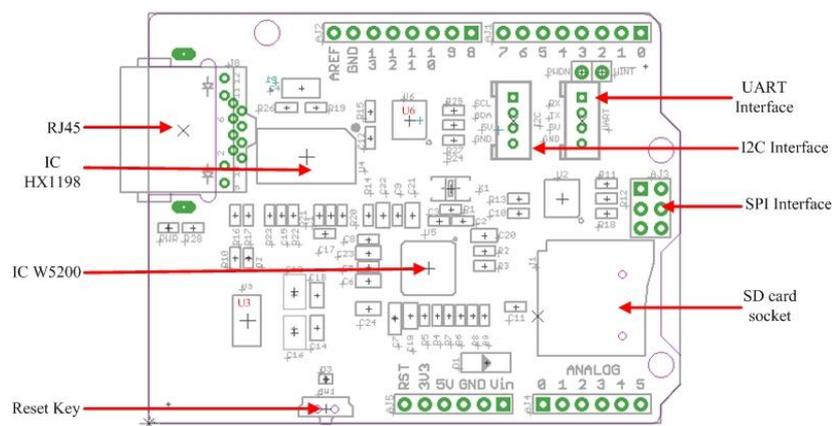


Figura 3-8 – Componentes do Ethernet Shield

A figura 3-8 mostra com mais detalhes os principais componentes do Shield Ethernet. É possível notar na figura o conector padrão RJ-45 para que seja permitido o encaixe do cabo de rede entre o módulo Ethernet e o roteador de rede local assim como o micro-SD mencionado. O micro-SD presente neste componente é utilizado quando qualquer tipo de arquivo for necessário estar presente na página Web, como por exemplo uma imagem.

O controlador W5200 é o responsável por criar uma conexão entre a plataforma Arduino com a Internet ou outros dispositivos em uma rede local. Este chip suporta protocolos da camada de transporte do modelo OSI, tornando possível o envio de pacotes de forma segura. Como mencionado anteriormente, este chip se conecta à plataforma Arduino através do SPI com

uma velocidade de até 80MHz. A tensão de alimentação suportada por este controlador é de 3.3V até 5V o que satisfaz as necessidades do sistema proposto.

3.4 Desenvolvimento do Software

O desenvolvimento do software do sistema de monitoramento remoto é realizado através da plataforma de desenvolvimento de software do Arduino. A linguagem utilizada é baseada em C/C++ e a passagem do código desenvolvido para a plataforma Arduino é feita através da porta USB. A seguir, os códigos desenvolvidos são mostrados em partes, porém trata-se apenas de um software desenvolvido para todo o sistema de monitoramento remoto.

O código 3-1 é utilizado para processamento das informações fornecidas pelo sensor LM35. O LM35 é declarado na porta analógica A4 da plataforma Arduino, pois na montagem do circuito esta porta é utilizada para ler os dados coletados pelo sensor. No valor de "T" é utilizado um comando para leitura do LM35 e também uma regra de três simples, visto que o Arduino lê valores inteiros entre 0 e 1023 conforme referenciado no capítulo 2 e varia 10mV para cada grau de temperatura em Celsius.

O valor 1023 representa 500 °C, sendo necessária a regra de três para converter o valor analógico lido em graus Celsius.

```
const int LM35 = A3;

int D= 0;

float T = 0;

void setup() {

  Serial.begin(9600);

}
```

```

void loop() {

  T = 0;

  D = 0;

  D = analogRead(LM35);

  T = (float) (D * 0.4887585532746823069403714565); //conversão para
  graus Celsius – (D*500)/(1023)

  Serial.println(D);

  Serial.print("Temperatura (C): ");

  Serial.println(T);

  delay(1000);

}

```

**Código 3.1 – Detectando o resultado da temperatura
(Fonte: Autor)**

O código 3-2 é utilizado para processamento das informações fornecidas pelo sensor ACS712. A variável “L” determina o valor lido pelo sensor de corrente, que após conversão é encontrado o valor em Ampère da corrente elétrica. A variável “L” também corresponde ao valor lido em mV, já que a plataforma Arduino faz leituras desta forma, sendo necessário o cálculo indicado no código abaixo para se encontrar o resultado da corrente elétrica em Àmpere.

```

const int corrente = A5; //Porta análogica do Arduino

int mVporA = 66; //66mv para cada àmpere

valor_inicial = 0;

int acs_offset = 2500;

```

```
double V = 0;

double A = 0;

void setup(){

Serial.begin(9600);

}

void loop(){

valor_inicial = analogRead(corrente);

V = (valor_inicial / 1023) * 5000; // Cálculo da tensão em mV do sensor
ACS712

A = (((V - acs_offset) / mVporA));

Serial.print("A = "); //Indica a corrente medida após a conversão
necessária

Serial.println(A,2); // é utilizado também 2 casas decimais no resultado

delay(2500);

}
```

**Código 3.2 – Detectando o resultado da corrente
(Fonte: Autor)**

Para a transmissão dos dados coletados pelos códigos acima, é desenvolvido um código com informações sobre o IP (Internet Protocol) e adicionado dados sobre a página HTML. O código HTML é desenvolvido também na plataforma IDE do Arduino, conforme mencionado no capítulo 2 deste trabalho.

A página HTML pode ser alterada de forma rápida e eficiente, como por exemplo mudanças de fontes e cores do texto. O exemplo a seguir mostra parte do código desenvolvido para a página HTML que é disponibilizada via

interface Web para monitoramento remoto dos dados coletados pelos sensores.

Já o código 3-3 mostra o cálculo necessário para encontrar o consumo de energia elétrica de um equipamento considerando que este equipamento irá permanecer ligado na rede elétrica durante 24 horas diárias.

```
float P = (emon1.Vrms) * (A); //Produto da tensão versus corrente
float Consumo = (P * 24)/1000; //Cálculo em kW/dia do consumo de
energia
```

**Código 3.3 – Cálculo do consumo de energia elétrica diário
(Fonte: Autor)**

O código 3-4 mostra o início da página HTML, as declarações são necessárias para indicar que trata-se de uma página HTML utilizando o protocolo HTTP/1.1, que é um protocolo de comunicação utilizado para troca e transferências de informações entre o HTML e o HTTP.

```
client.println("<html>"); //INICIO PAGINA HTML
client.print("<head><title>UniCeub</title>"); //HEAD DO HTML E
ESTILOS
client.print("<style>");
client.print("table { border-collapse: collapse; width:80%; background-
color:#fff; text-align: left; margin-left: 10%; }"); //ESTILO TABELA
client.print("table tr { border-top: 1px solid black; }"); //ESTILO LINHA
TABELA
client.print("body { text-align: center; }"); //ESTILO BODY
```

```

    client.print(".container { margin-left: 15%; margin-right: 15%; }");
//ESTILO DIV CLASS CONTAINER

    client.print("h1 { color: #104E8B; font-size: 32px; }"); //ESTILO H1

    client.print("h2 { color: #104E8B; font-size: 18px; margin-bottom:
40px; }"); //ESTILO H2

    client.print("h3 { color: #101010; font-size: 15px; margin: 0px; }");
//ESTILO H3

    client.print("</style></head><body><div class='container'>"); //FIM
DO HEAD E ESTILOS INICIO DA BODY

    client.print("<h1>UniCeub - Centro Universitario de Brasilia</h1>");

    client.print("<div class='container'>");

    client.print("<h3>Curso:                Engenharia                da
Computacao</h3><h3>Nome:  Mateus  de  Melo  Sobral</h3><h3>Ra:
20950833</h3></div>");

    client.print("<h2>Sistema de Monitoramento Remoto de Consumo de
Energia e Temperatura</h2>");

    client.print("<div class='container'><table>");

    client.print("<tr><td>Temperatura (C) :</td><td>");

    client.print(T);

    client.print("</td></tr>");

    client.print("<tr><td>Tensao Eletrica (V)</td><td>");

    client.print(emonl.Vrms);

    client.print("</td></tr>");

    client.print("<tr><td>Corrente Eletrica (A) </td><td>");

    client.print(A,2);

```

```
client.print("</td></tr>");  
  
client.print("<tr><td>Consumo de Energia (kW/dia)</td><td>");  
  
client.print(Consumo);  
  
client.print("</td></tr>");  
  
client.print("</table></div>");  
  
client.print("</div></body></html>"); //FIM DO HTML
```

Código 3.4 – Página HTML
(Fonte: Autor)

O título declarado na penúltima linha de parte deste código mostra o nome que é exibido no navegador de Internet ao se conectar com a página HTML, podendo ser facilmente alterado para qualquer outro nome. A fonte do título é mostrada no início do código e também pode ser alterada de acordo com preferências do usuário.

3.5 Rede local

No sistema proposto é utilizado um roteador DI-624 da fabricante D-Link com o objetivo de captar as informações recebidas do Arduino e do Shield Ethernet através do cabo de rede RJ-45 e em seguida, fornecer estes dados para a estação de monitoramento em uma rede Wi-fi local, com um limite de até quatro acessos simultâneos.

O acesso para monitoramento das informações coletadas é realizado através um navegador de internet. É necessário inserir o endereço IP da plataforma Arduino e em seguida a página HTML é exibida para o usuário onde é possível visualizar por meio da interface web, os dados necessários para monitoramento remoto.

O código 3-5 mostra as definições de rede que são necessárias para tornar a plataforma Arduino em um servidor. As configurações são feitas após o

acesso ao roteador DI-624 para coletar os dados sobre a máscara de rede, gateway e IP.

```
//Definicoes de IP, mascara de rede e gateway  
  
byte mac[] = {  
  
    0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };  
  
IPAddress ip(192,168,0,101);    //Define o endereco IP  
  
IPAddress gateway(192,168,0,1); //Define o gateway  
  
IPAddress subnet(255, 255, 255, 0); //Define a máscara de rede  
  
EthernetServer server(80); //Inicializa o servidor web na porta 80
```

**Código 3.5 – Configuração da rede local e inicialização do servidor web
(Fonte: Autor)**

A figura 3-9 detalha as configurações de rede do Arduino para acesso à página web.

Através destas configurações, é possível tornar a plataforma Arduino juntamente com o Shield Ethernet em um servidor de dados, onde contém armazenada todas as informações que são coletadas pelos sensores na estação de coleta. Assim, é possível acessar os dados de uma estação de monitoramento remota apenas se conectando na rede local roteada pelo D-Link e em seguida entrar com o IP da plataforma Arduino em um navegador qualquer para acessar a página HTML.

Endereço MAC	0xDE:0xAD:0xBE:0xEF:0xFE:0xED
Endereço de IP	192.168.0.101
Máscara de Sub-Rede	255.255.255.0
Endereço do Gateway	192.168.0.1

**Figura 3-9 – Tabela com informações de rede do Arduino
(Fonte: Autor)**

3.6 Sensor de tensão

Para medir a tensão elétrica é construído um sensor de tensão utilizando um transformador 220V/9V assim como 3 resistores de 10K Ω e um resistor de 100K Ω , além de um capacitor de 10 μ conforme esquemático desenvolvido na figura 3-10. Estes resistores e capacitor são utilizados para diminuição da corrente elétrica de saída do transformador até a entrada na plataforma Arduino.

Já o transformador é utilizado para conversão da tensão de entrada, seja ela 110V ou 220V para 9V, que já é uma tensão elétrica aceitável para utilização da plataforma Arduino. Pois a tensão de entrada recomendada para o Arduino é entre 7V e 12V conforme mencionado anteriormente neste projeto. Já o circuito montado é necessário para redução da tensão de saída do transformador que é de 9V, com isso, o circuito é desenvolvido para diminuir esta tensão para em seguida ser possível realizar a leitura da tensão elétrica através do processamento do Arduino.

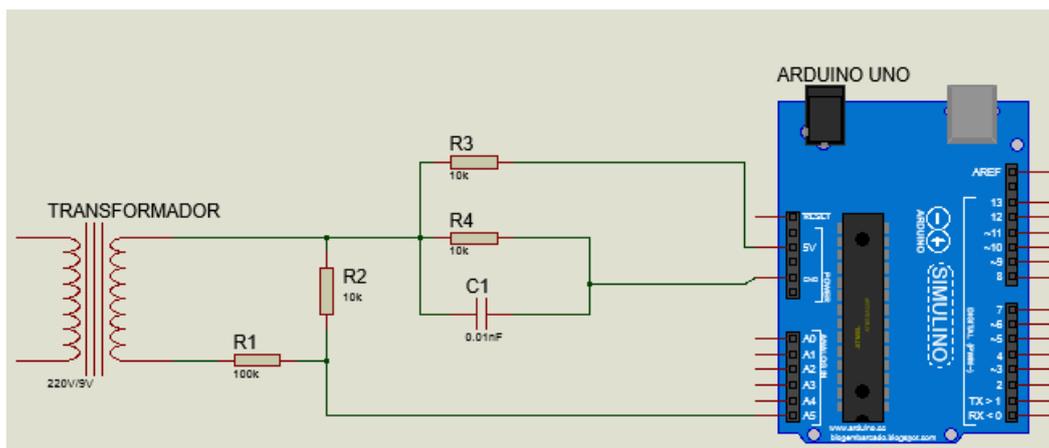


Figura 3-10 – Circuito do sensor de tensão
(Fonte: Autor)

Para a medição da tensão, foi baixada e instalada a biblioteca EmonLib.h que é de fácil instalação no IDE do Arduino. A biblioteca EmonLib.h é uma biblioteca especificamente desenvolvida para atender demandas de coletas de informações sobre a corrente elétrica, tensão elétrica e consequentemente o consumo de energia, fornecido pela potência elétrica que é o produto da corrente elétrica com a tensão da rede elétrica conforme mencionado anteriormente neste trabalho.

```
#include "EmonLib.h"

EnergyMonitor emon1;

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

  emon1.voltage(2, 236, 1.7);
}

void loop ()
{
```

```
emon1.calcVI(20,2000);  
  
float supplyVoltage = emon1.Vrms;  
  
Serial.print("Tensao eletrica (V): ");  
  
Serial.println(emon1.Vrms);  
  
}
```

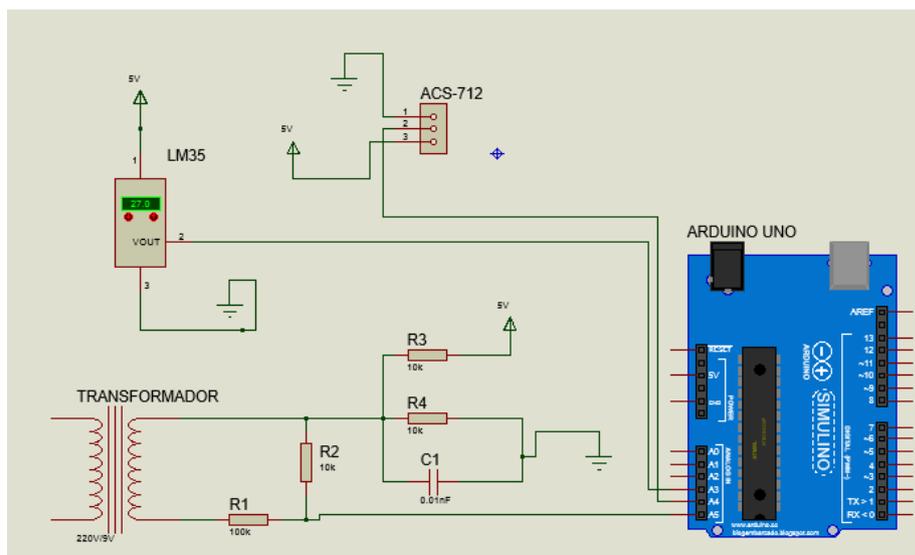
Código 3.6 – Declaração da EmonLib.h
(Fonte: Autor)

O código 3-5 mostra o código desenvolvido para a coleta da tensão elétrica da rede. Para o sistema proposto, o código é calibrado para coletar tensões alternadas da rede elétrica de 220V, porém basta alterar o segundo valor no parâmetro “emon1.voltage” caso o desejo de medição seja em uma rede elétrica de 127V.

É necessário também criar uma instância de medição da tensão, esta instância é criada logo após a declaração da biblioteca EmonLib no código.

3.7 Placa do circuito impresso

O circuito completo do sistema de monitoramento de consumo de energia e temperatura é mostrado na figura 3-11 que contém todos os componentes utilizados para montagem do protótipo em uma placa de cobre.



**Figura 3-11 – Esquemático elétrico do circuito proposto
(Fonte: Autor)**

Após o desenvolvimento do esquemático elétrico do circuito final, é desenvolvido o *lay-out* conforme a figura 3-12 para impressão e montagem do protótipo em uma placa de cobre com todos os componentes necessários para o funcionamento do sistema de monitoramento proposto.

O *layout* do sistema de monitoramento proposto é mostrado na figura 3-16 e contém os dois sensores ligados por trilhas de conexão até os seus devidos conectores, seja ele de alimentação ou aterramento, tanto a alimentação quanto o aterramento são fornecidos pela plataforma Arduino.

As trilhas da placa de cobre servem para realizar o papel dos jumpers que são utilizados no primeiro do projeto. Nelas, a alimentação dos sensores é feita de maneira eficiente, assim como o aterramento para que os sensores possam funcionar de maneira correta e eficiente.

medição de tensão são fixados na placa de cobre conforme esquemático elétrico e *layout* da placa apresentados anteriormente.

O protótipo final é capaz de garantir a coleta das informações necessárias para o monitoramento destes dados através da Web conforme a proposta apresentada. A figura 3-13 mostra o protótipo final montado e operacional para ser utilizado no monitoramento de consumo de energia e temperatura de racks em data center.

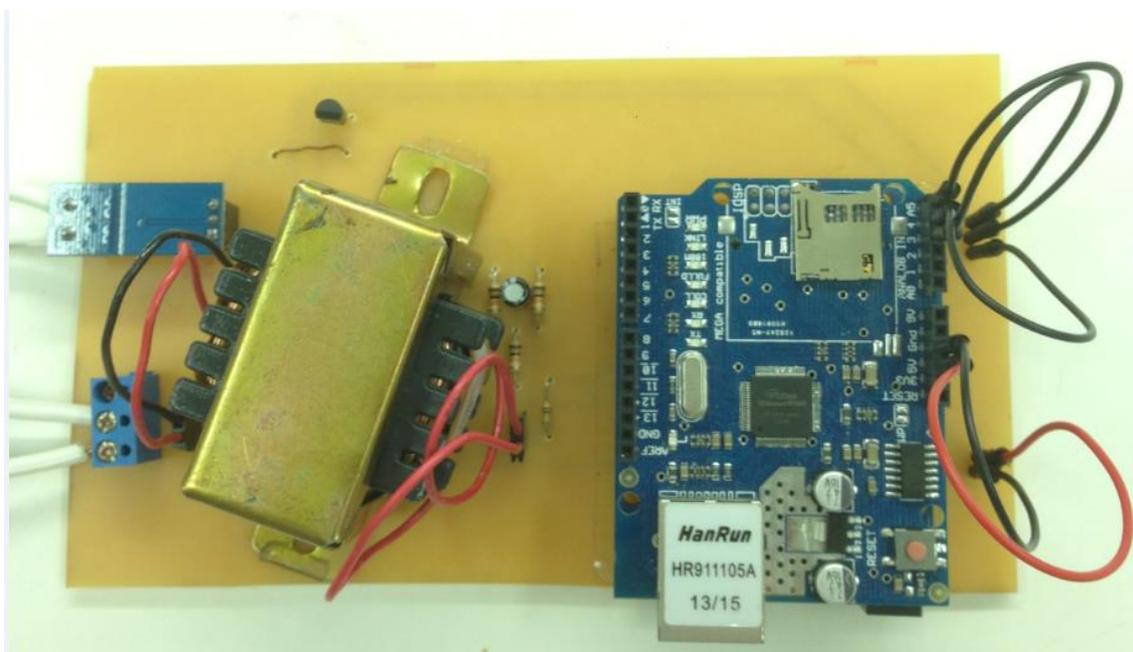
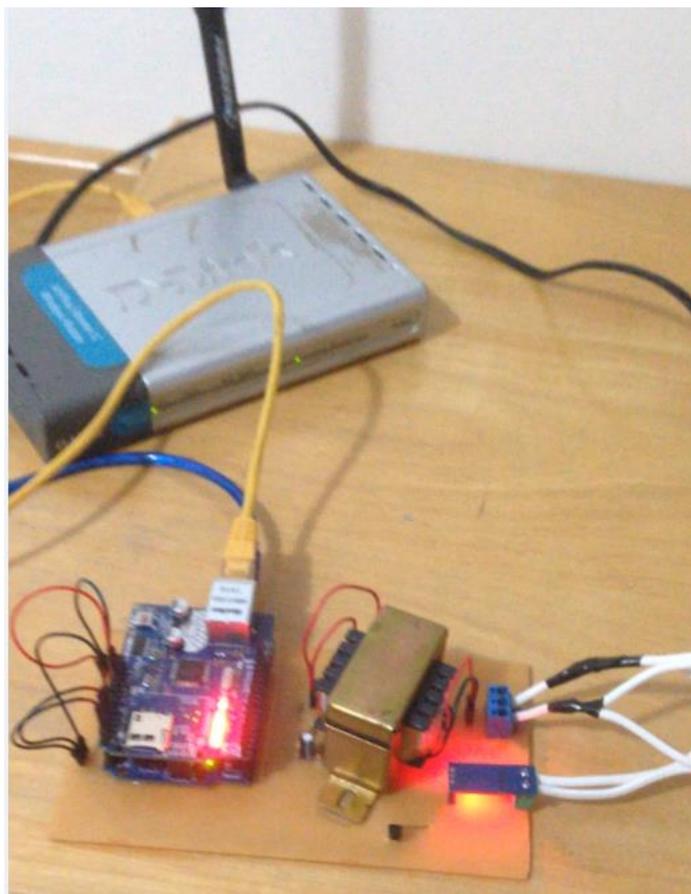


Figura 3-13 – Protótipo final
(Fonte: Autor)

O protótipo final mostra também exatamente os componentes elétricos que são projetados e soldados na placa de cobre conforme item anterior deste projeto. No canto esquerdo é possível visualizar a entrada da tensão elétrica, que em seguida passa por uma conversão, realizada pelo transformador, para finalmente chegar no circuito desenvolvido para a coleta da tensão. O circuito é composto de resistores para diminuir a tensão elétrica, já que a plataforma Arduino suporta uma tensão de 5V conforme descrito no referencial teórico. Já o capacitor deste circuito é utilizado para evitar qualquer tipo de ruído ou interferência.

Acima da entrada de tensão, é possível também visualizar o sensor ACS712 que é responsável pela coleta da corrente elétrica do equipamento e também o sensor de temperatura. Todas as ligações entre estes dispositivos elétricos estão embaixo da placa e é possível visualizar todas as conexões através da placa de cobre desenvolvido no item anterior deste trabalho.



**Figura 3-14 – Protótipo final com Roteador de rede
(Fonte: Autor)**

A figura 3-14 mostra o protótipo final no momento de um teste já ligado com o roteador através de um cabo padrão RJ-45. É possível visualizar também todos os sensores conforme mencionado na figura anterior.

CAPÍTULO 4 TESTES E RESULTADOS

Neste capítulo são apresentados os detalhes dos testes realizados de todos os equipamentos usados no projeto, com o objetivo de analisar o desempenho e avaliar os resultados conforme solução proposta. Todos os sensores são testados, assim como a estação de monitoramento para garantir que a informação para monitoramento seja recebida de forma adequada e correta.

4.1 Cenários de testes

4.1.1 Cenário 1

O objetivo deste primeiro cenário é garantir o funcionamento correto do sensor de temperatura LM35. Um ferro de solda é encostado no sensor para testá-lo. Após 2 segundos já é possível perceber a alteração da temperatura através do display do software da plataforma Arduino. A tensão de saída do LM35 varia 10mV para cada grau Celsius.

A figura 4-1 mostra o desempenho do sensor LM35, são realizadas 8 coletas de temperatura assim como a tensão em milli Volts que é lida pela plataforma Arduino.

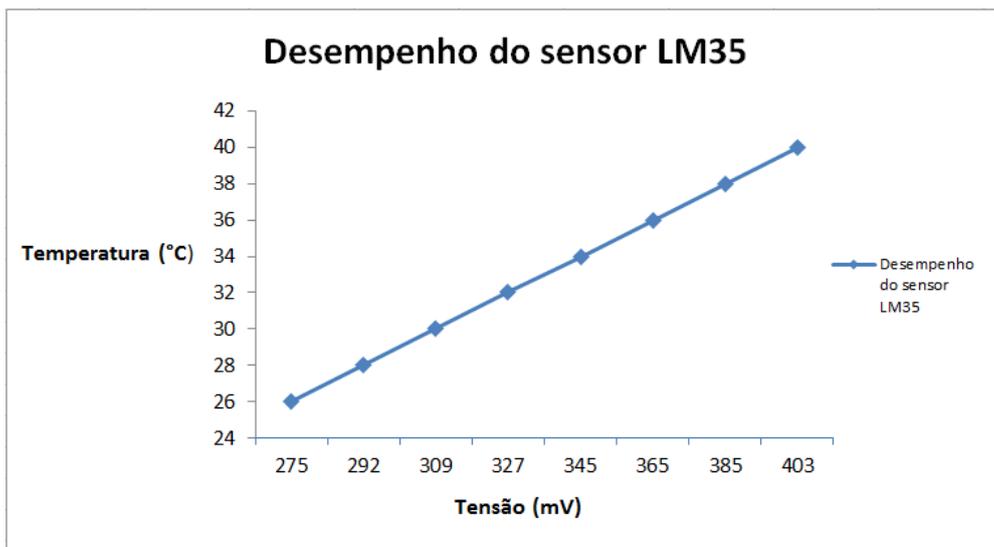


Figura 4-1 – Desempenho do sensor LM35
(Fonte: Autor)

Conforme o referencial teórico presente no capítulo 2 deste trabalho, a margem de erro do LM35 é de $3/4^{\circ}\text{C}$ para mais ou para menos, ou seja, 1mV representa cerca de $1/10^{\circ}\text{C}$, com isto, é possível concluir que a margem de erro está de acordo com a especificação do fabricante do sensor de temperatura. Também é testado essa variação de tensão de saída através da porta analógica do arduino e também é possível concluir que a variação ocorre de forma correta garantido o perfeito funcionamento do LM35 para o projeto.



Figura 4-2 – Teste do LM35 com o Clima Tempo
(Fonte: Autor)

Já a figura 4-2 indica uma comparação feita com o website Clima Tempo que é responsável por fornecer informações sobre a previsão do tempo

de diversos lugares em todo o Brasil. Ao lado esquerdo, é possível visualizar a temperatura fornecida no Clima Tempo no momento do teste realizado. Ao lado direito é mostrada a temperatura coletada pelo sensor de temperatura utilizado neste projeto.

4.1.2 Cenário 2

O objetivo deste cenário é garantir o correto funcionamento do sensor de corrente ACS712. O teste realizado para avaliar o funcionamento deste sensor é através de uma tomada fêmea ligada ao sistema de monitoramento. Nesta tomada é ligado um ferro elétrico com uma potência de 1000 W, assim como uma alimentação da rede elétrica de 220V.

Com isso, através da equação do cálculo da potência é possível identificar a corrente elétrica do ferro. Se for realizada a divisão da potência pela tensão ($1000/220$) o resultado será uma corrente por volta de 4,5A. A figura 4-3 mostra os resultados coletados deste teste.

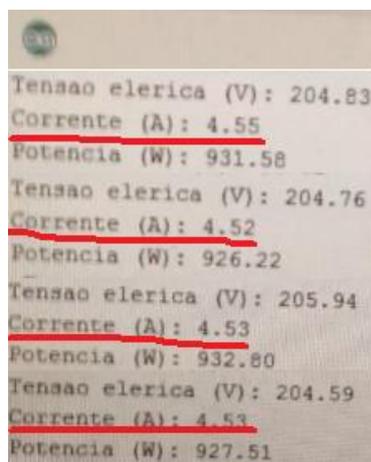


Figura 4-3 – Teste do ACS712
(Fonte: Autor)

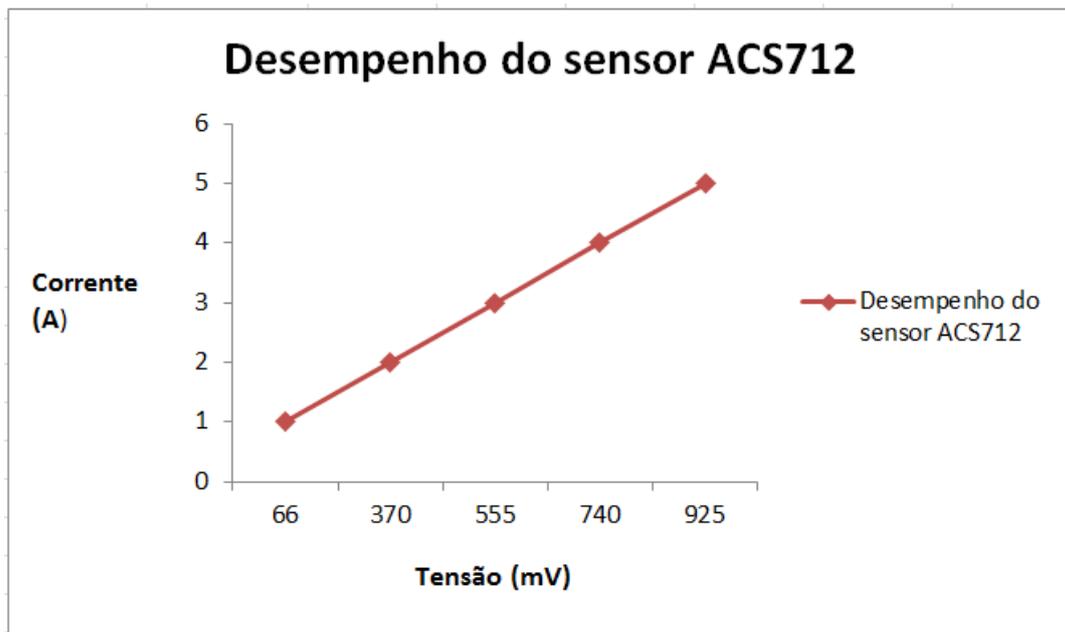


Figura 4-4 – Desempenho do ACS712
(Fonte: Autor)

Já a figura 4-4 mostra o gráfico que sinaliza o desempenho do sensor ACS712. Para este sensor, conforme mencionando no capítulo 2, a sua sensibilidade é de 66mV por ampere. Desta forma, o gráfico mostra a variação da conversão de ampères para cada intervalo de 66mV por coleta realizada.

O sensor não sofre grandes variações de desempenho para as temperaturas ambientes em que foi testado.

4.1.1 Cenário 3

O objetivo deste terceiro cenário é avaliar o recebimento das informações coletadas pelo sensor de tensão que foi construído. O transformador de tensão é ligado em uma tomada simples de 220V. Os resultados coletados são mostrados na figura 4-5, juntamente com os resultados coletados por um multímetro que é utilizado para comparação da real tensão que está sendo coletada pelo circuito e código desenvolvido.

Neste cenário, também é construído um gráfico de comparação entre as medições realizadas pelo multímetro e pelo circuito desenvolvido. Para

finalizar, é utilizado um osciloscópio para avaliar a qualidade do sinal de saída do transformador.

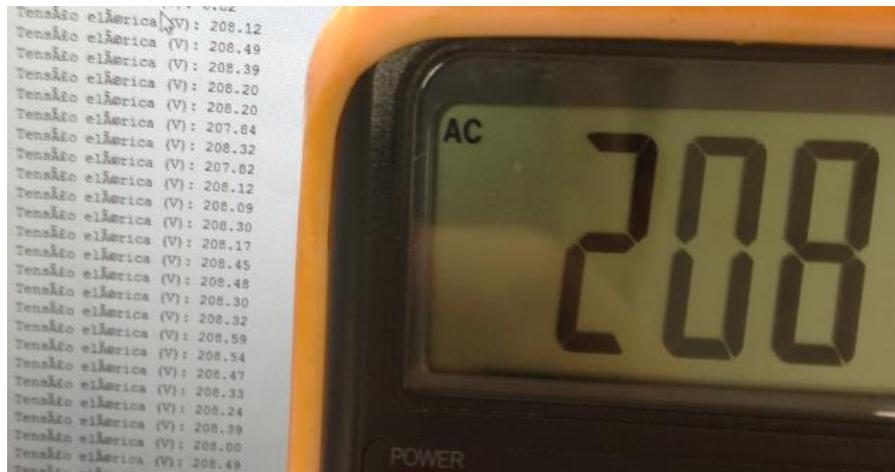


Figura 4-5 – Medição da tensão elétrica pelo sensor desenvolvido e pelo multímetro (Fonte: Autor)

Ao lado direito, o multímetro fornece uma tensão de 208V e ao lado esquerdo é possível visualizar que o circuito desenvolvido coleta uma tensão elétrica entre 207V – 208V o que garante a confiabilidade do sistema proposto.

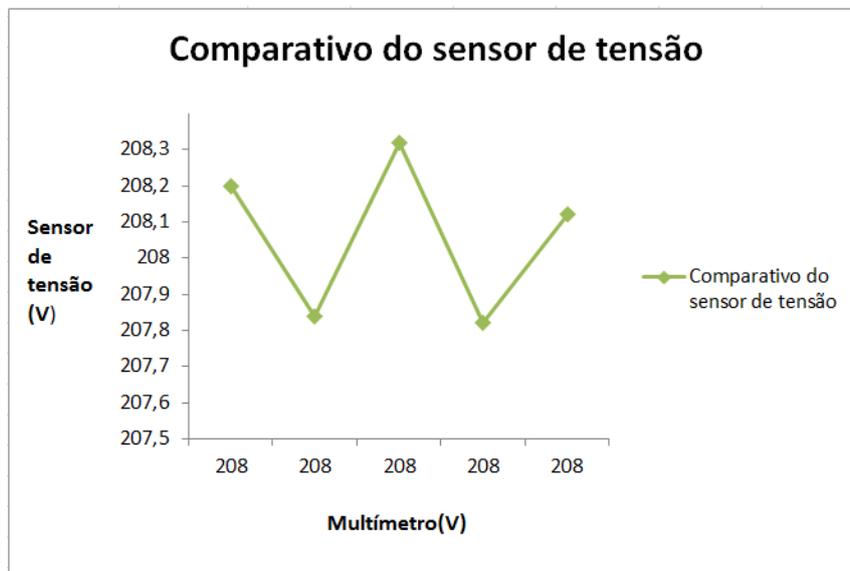


Figura 4-6 – Comparativo do sensor de tensão x multímetro (Fonte: Autor)

Já a figura 4-6 mostra um comparativo entre 5 medições realizadas pelo sensor de tensão desenvolvido e 5 medições que foram feitas pelo multímetro. O resultado é mostrado em forma de gráfico de linhas onde é possível perceber que a variação da tensão está de acordo com o sistema proposto. A diferença de medição entre o circuito desenvolvido e o multímetro é pequena, o que não afeta o resultado do projeto proposto.

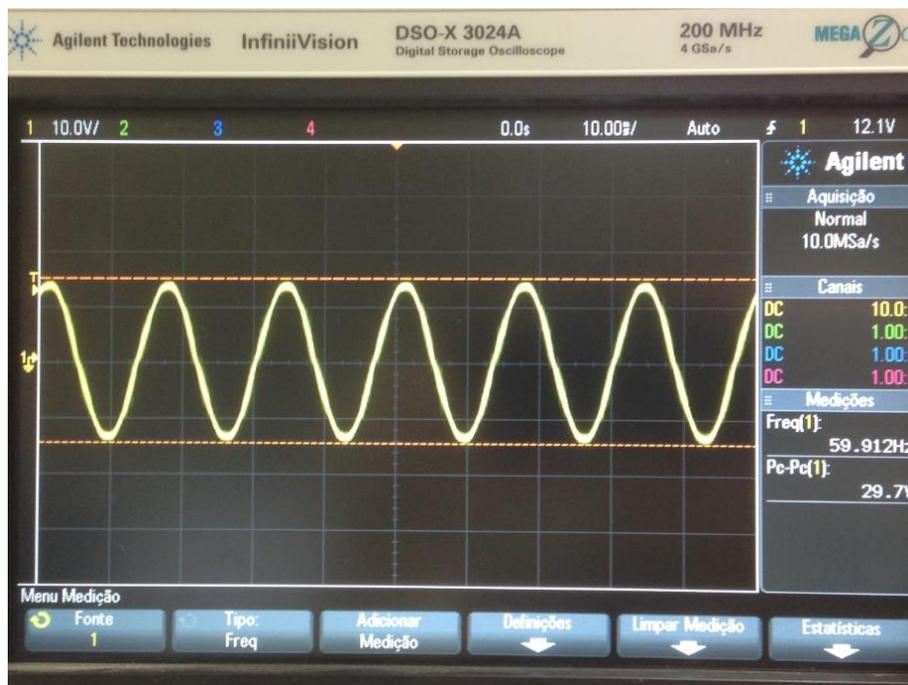


Figura 4-7 – Osciloscópio utilizado para verificar o sinal de saída do transformador (Fonte: Autor)

A figura 4-7 mostra o resultado da saída do transformador, que conforme descrito anteriormente é de 9V. Este sinal é verificado para garantir que a qualidade da saída da tensão esteja correta, assim como a presença de possíveis ruídos para que em seguida o circuito composto de resistores e capacitor possa receber o sinal para finalmente ser enviado para a plataforma Arduino realizar a leitura da tensão da rede elétrica, já que a plataforma Arduino opera à uma tensão de 5V.

4.1.2 Cenário 4

O objetivo deste quarto cenário é avaliar a potência do equipamento que foi testado, para verificar a veracidade e confiabilidade do sistema proposto. Neste caso, o mesmo ferro elétrico é utilizado para ser possível verificar a potência elétrica do mesmo através do produto da tensão elétrica calculada com a corrente elétrica coletada pelo sensor ACS712.

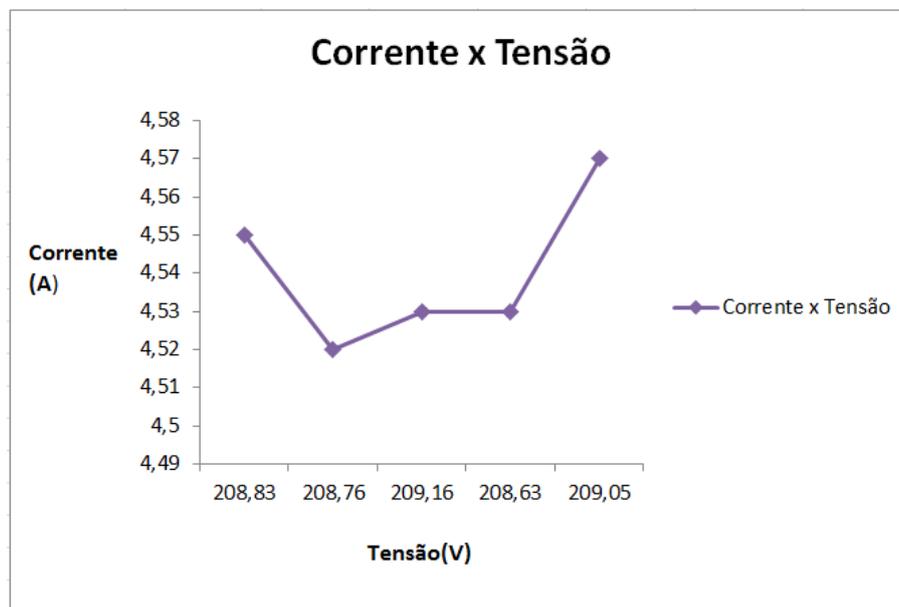


Figura 4-8 – Resultado da coleta da corrente do ferro elétrico e tensão elétrica da rede (Fonte: Autor)

A figura 4-8 mostra o gráfico da corrente elétrica com a tensão elétrica que são coletados através dos sensores de corrente e tensão respectivamente. A potência do ferro elétrico utilizado é de 1000 W e são coletadas 5 amostras para o teste.

Todas as amostras fornecem uma potência entre 947 W e 955 W garantindo assim o correto funcionamento do sistema de coleta de dados para o monitoramento do consumo de energia. A variação ocorre devido a tensão elétrica ser alternada e não ser exatamente 220V, assim como a corrente elétrica do ferro que também é alternada.

4.1.3 Cenário 5

O objetivo deste último cenário é avaliar o recebimento das informações coletadas pelos sensores através da estação de monitoramento remota em rede local. É realizado um acesso através do endereço 192.168.0.101 que corresponde ao endereço IP da plataforma Arduino.

A página HTML foi aberta com sucesso conforme detalhado no capítulo 3. A página web é desenvolvida e programada para ser atualizada a cada 5 segundos, por isso, a chegada as novas informações tendem a sofrer com este pequeno intervalo de atualização da página web.



**Figura 4-9 – Protótipo coletando e enviando dados para a página web
(Fonte: Autor)**

A figura 4-9 exibe o cenário em que este teste foi realizado. Um ventilador de ar de 70W e 0,3A foi conectado ao sistema de monitoramento remoto para ser verificada a confiabilidade das informações coletadas via interface web desenvolvida. É possível também visualizar todo o sistema de monitoramento em operação, desde a plataforma Arduino conectada no

roteador através de um cabo RJ-45 via Shield Ethernet até os sensores mencionados durante este trabalho.

UniCeub - Centro Universitario de Brasilia

Curso: Engenharia da Computacao
Nome: Mateus de Melo Sobral
Ra: 20950833

Sistema de Monitoramento Remoto de Consumo de Energia e Temperatura

Temperatura (C) :	24.01
Tensao Eletrica (V)	211.07
Corrente Eletrica (A)	5.38
Consumo de Energia por dia (kW/dia)	27.25

**Figura 4-10 – Página HTML com os dados coletados
(Fonte: Autor)**

Já a figura 4-10 mostra a página HTML em funcionamento com as informações coletadas pelos sensores de um ferro elétrico de 5A e 1100W garantindo a confiabilidade do sistema visto que a corrente elétrica mostrada é de 5,38A e o consumo de energia elétrica é calculado por dia, no caso de 24 horas diárias o ferro iria consumir 27,25 kW/dia.

Para acessar a página Web, é necessário conectar na rede local com o nome SMR (Sistema de Monitoramento Remoto) que é roteada pelo roteador utilizado também no projeto.

Desta forma, neste cenário é testado a conexão com a rede local e em seguida é acessado o IP, configurado no capítulo 3, da plataforma Arduino através de um navegador web. A página web é atualizada a cada 5 segundos para garantir que qualquer mudança na coleta dos dados seja atualizada rapidamente na página de monitoramento remoto. O monitoramento remoto pode ser realizado também por tablets ou smartphones.

CAPÍTULO 5 CONCLUSÃO

De acordo com a alta demanda de equipamentos robustos para suprir as necessidades da sociedade mundial na indústria da Tecnologia da Informação, identificou-se a necessidade também de monitoramento destes equipamentos para, em paralelo com o conceito de TI verde, auxiliar o meio ambiente provendo informações importantes sobre o consumo de energia e temperatura destes equipamentos utilizados no meio da Tecnologia da Informação. A necessidade de soluções tecnológicas para ajudar o meio ambiente e a promover a sustentabilidade global é uma tendência, e por isso foi proposto este sistema de monitoramento em tempo real.

Com isso, este projeto teve como principal foco o desenvolvimento de um protótipo que é responsável por monitorar as informações do consumo de energia e temperatura destes equipamentos da área de TI e Telecomunicações afim de auxiliar o uso da tecnologia em prol da tecnologia.

O objetivo foi cumprido de forma adequada. Foi desenvolvido e construído um sistema de monitoramento de consumo de energia e temperatura para equipamentos de TI de fácil instalação e configuração. Os resultados obtidos foram de alto desempenho e foram alcançados de forma satisfatória por não ultrapassarem uma margem de erro de 3%, além dos testes realizados comprovarem o correto funcionamento do sistema de monitoramento destas informações.

O principal foco do trabalho são os racks utilizados em datacenter tanto em empresas de Tecnologia da Informação, quanto no ramo da Telecomunicação. Por serem equipamentos robustos, e de acordo com a preocupação ambiental que vivemos nos últimos anos, identificou-se a necessidade de utilização de soluções tecnológicas para auxiliar o monitoramento do consumo de energia e temperatura destes equipamentos utilizados em larga escala em todo o mundo.

Na busca de minimizar as consequências geradas pelo alto consumo de energia ao meio ambiente, o projeto desenvolvido contribui para a sustentabilidade alertando e apontando para as empresas, através do sistema de monitoramento, os equipamentos que consomem energia em excesso, além da temperatura coletada do ambiente em que estes equipamentos estão instalados.

A construção do protótipo foi realizada com os conhecimentos absorvidos ao longo do curso de Engenharia da Computação que foram de extrema importância para o desenvolvimento do protótipo. Os conceitos aplicados neste trabalho vão desde a linguagem de programação e física até circuitos eletrônicos, sistemas de comunicação e redes de computadores, incluindo aqueles que foram estudados no referencial teórico presente capítulo 2 deste trabalho.

A linguagem de programação foi importante para o desenvolvimento de todo o software necessário para coletar as informações, assim como a página de monitoramento que foi desenvolvida na linguagem HTML. Já os circuitos eletrônicos foram de extrema importância para construção de todo o protótipo desenvolvido neste trabalho. Redes de computadores e sistemas de comunicação contribuíram para o envio das informações coletadas para uma interface web, onde foi necessário a configuração de máscaras de redes e IP. E por fim, os conceitos da física foram aplicados para os cálculos necessários do consumo de energia através da corrente e tensão elétrica.

5.1 SUGESTÕES PARA TRABALHOS FUTUROS

Como sugestão de trabalhos futuros, é possível desenvolver um software com maior capacidade e integrado com banco de dados para armazenar informações diárias dos equipamentos que são utilizados dentro de uma empresa de Tecnologia da Informação ou Telecomunicações.

Para transmissão dos dados coletados pela estação de coleta, é possível utilizar outras formas de comunicação entre a estação de coleta e a estação de monitoramento, como por exemplo o envio destas informações através da tecnologia 3G ou 4G, que seria possível acessar o sistema de monitoramento não apenas de uma rede local, mas em qualquer máquina, ou telefone que possua um acesso à Internet.

Estas sugestões propostas irão contribuir para a realização de melhorias no sistema proposto, assim como novos estudos neste meio para aprimorar as tecnologias utilizadas em favor da própria tecnologia e desta forma ser possível contribuir para a sustentabilidade e meio ambiente.

REFERÊNCIAS

- alldatasheet.com*. (2014). Acesso em 19 de agosto de 2015, disponível em <http://www.alldatasheet.com/datasheet-pdf/pdf/168326/ALLEGRO/ACS712.html?>
- Allegro MicroSystems LLC. (2013). *Fully Integrated, Hall Effect-Based Linear Current Sensor IC*.
- ARDUINO.CC. (s.d.). Acesso em 15 de Agosto de 2015, disponível em <http://arduino.cc.com>
- Cruz, E. A. (11 de 2014). *ELETRICIDADE BÁSICA - CIRCUITOS EM CORRENTE CONTÍNUA*. ERICA.
- helpdev.com.br*. (2014). Acesso em 12 de agosto de 2015, disponível em <http://helpdev.com.br/2014/01/01/arduino-sensor-de-corrente-ac-712-em-ac-e-dc-monitoring-current-with-arduino-and-ac-712/>
- INMETRO/PBE. (2015). *Folder PBE*. Acesso em 18 de Agosto de 2015, disponível em [Inmetro: inmetro.gov.br](http://inmetro.gov.br)
- Jones, D. (2010). TI verde: Sustentabilidade - Sua empresa se importa com ela? *TechNet Magazine - Microsoft*.
- ladyada.net*. (2012). Acesso em 15 de setembro de 2015, disponível em <http://www.ladyada.net/learn/sensors>
- Makower, J. (2009). *A economia verde*. São Paulo: Editora Gente.
- McRoberts, M. (2011). *Arduino Básico*. São Paulo, SP: Novatec.
- Monk, S. (2013). *Programação com arduino - começando com sketches*. (Bookman, Ed.) Porto Alegre.
- National Semiconductor. (s.d.). *LM35 Datasheet*. Acesso em 20 de Agosto de 2015, disponível em www.alldatasheet.com
- Rezende, D., & Abreu, A. (2013). *Tecnologia da Informação aplicada a Sistemas de Informação Empresariais*. São Paulo: Atlas.
- Richter, R. (2012). *TI Verde: Sustentabilidade por meio da Computação em Nuvem*. São Paulo.
- TEXAS INSTRUMENTS. (2013). Acesso em 15 de setembro de 2015, disponível em <http://www.ti.com/lit/ds/symlink/lm35.pdf>
- Torres, G. (2014). *Redes de Computadores: Versão Revisada e Atualizada*. São Paulo, SP: Novaterra.

waihung.net. (s.d.). Acesso em 12 de agosto de 2015, disponível em waihung.net/arduino-ethernet-shield-r3

Westleya, G. (2009). Commercial electricity demand in a Central American economy.