

FACULDADE DE TECNOLOGIA SÃO PAULO – FATEC

DEPARTAMENTO DE SISTEMAS ELETRÔNICOS
ELETRÔNICA INDUSTRIAL

SISTEMA DE CONTROLE DE AMBIENTES - SCA

MILTON LUIZ GUETTI JUNIOR
THIAGO NOGUEIRA DE CHAVES
THIAGO REZENDE DA SILVA

SÃO PAULO - SP

2021



MILTON LUIZ GUETTI JUNIOR
THIAGO NOGUEIRA DE CHAVES
THIAGO REZENDE DA SILVA

SISTEMA DE CONTROLE DE AMBIENTES - SCA

Monografia apresentada à Faculdade de Tecnologia São Paulo, como parte dos requisitos para obtenção do título de Tecnólogo em Eletrônica Industrial.

Orientador: Prof. Me. Mauricio Antônio Deffert

São Paulo - SP

2021

Ficha Catalográfica

Guetti Junior, Milton; De Chaves, Thiago; Da Silva, Thiago.
SISTEMA DE CONTROLE DE AMBIENTES - SDA / Milton Luiz
Guetti Junior, Thiago Nogueira de Chaves, Thiago Rezende da
Silva – 2021.77p. Monografia – Curso Superior de Tecnologia
em Eletrônica Industrial – Faculdade de Tecnologia de São
Paulo, 2021. Orientador: Prof. Me. Maurício Antônio Deffert
1. Controlador de ambientes 2. Sensor LDR 3. Sensor DHT

MILTON LUIZ GUETTI JUNIOR
THIAGO NOGUEIRA DE CHAVES
THIAGO REZENDE DA SILVA

Sistema de controle de ambientes

Aprovada em: ___ / ___ / ___

Dr. Victor Sonnenberg

Coordenador do curso / Chefe do Departamento de Sistemas Eletrônicos

Me. Maurício Antônio Deffert
Orientador

Dr. Leonardo Frois Hernandez
Avaliador

Dr. Victor Sonnenberg
Avaliador

Agradecimentos

Gostaríamos de agradecer e dedicar este trabalho, primeiramente a todos nossos familiares, que participaram de todos os momentos no decorrer do curso, principalmente neste último ano que como todos sabemos tivemos muitos contratempos.

Agradecemos também aos nossos colegas de sala que desde o início foram essenciais para o desenvolvimento do aprendizado.

A instituição FATEC-SP, aos docentes, diretores, coordenadores e administração que proporcionaram o melhor dos ambientes para que esse trabalho fosse realizado.

Aos professores que ofereceram todo seu conhecimento e dedicação no ensino. Ao Prof. Maurício A. Deffert pela oportunidade e apoio durante todo o processo de construção desse TCC.

“A tarefa não é tanto ver aquilo que ninguém viu, mas pensar o que ninguém ainda pensou sobre aquilo que todo mundo vê”

Arthur Schopenhauer

RESUMO

Atualmente, as pessoas buscam cada vez mais conforto e comodidade, principalmente na realização de tarefas simples do cotidiano. Acender as luzes ou ligar uma cafeteira de forma automática em horários programados são típicos exemplos que podem fazer parte de uma nova rotina para as pessoas. Além disso, a utilização de sistemas de controle de ambiente se faz necessária e indispensável em diversos ambientes industriais, sendo a sua aplicabilidade utilizada desde a fabricação, armazenamento até o transporte dos produtos industrializados. Por meio de sensores que serão os olhos, ouvidos, olfato e até mesmo o tato é possível realizar essas atividades a distância, que são de um simples comando na tela de um celular por um aplicativo ou em uma página *web*. O desenvolvimento de um sistema microcontrolado de monitoramento e controle de ambientes que possa servir também como instrumento de medição de temperatura, umidade e luminosidade. O sistema visa permitir por meio do recurso *IoT* (Internet das coisas), gerenciar remotamente ambientes que necessitam de controle e monitoração. Existem várias aplicabilidades práticas para um sistema que possa monitorar e, se necessário, acionar mecanismos automatizados para correção dessas variáveis em ambientes controlados, contudo é na domótica que se observa uma maior predominância. O dispositivo irá fazer o registro em um banco de dados, das leituras periódicas efetuadas ao longo de um determinado espaço de tempo possibilitando a criação de tabelas e gráficos. Além de possuir uma IHM a qual poderemos interagir ajustando um intervalo de parâmetros para acionamento de alarmes ou comandos em caso de ultrapassar unidades ajustadas. O protótipo obteve êxito na leitura dos sensores e nos ajustes dos intervalos das unidades dos sensores, por meio de uma comunicação serial consegue enviar os dados a um serviço *IoT* que armazena os dados em uma plataforma *web* que podem ser acessados por qualquer dispositivo com acesso à internet por meio da plataforma de nuvem *ThinkSpeak*. O acionamento dos periféricos que atuarão no controle do ambiente em caráter de simulação, será representado por um conjunto de *led's*.

Palavras-chave: Controle de ambientes; Temperatura; Umidade; Luminosidade.

ABSTRACT

Nowadays, people seek more and more comfort and convenience, especially when performing simple everyday tasks. Turning on the lights or switching on a coffee machine automatically at scheduled times are typical examples that can be part of a new routine for people. In addition, the use of environmental control systems is necessary and indispensable in several industrial environments, being their applicability from manufacturing, storage to the transportation of industrialized products. Through sensors that will be the eyes, ears, smell and even touch it is possible to perform these activities at a distance, which are a simple command on a cell phone screen by an application or on a web page. The development of a microcontrolled environment monitoring and control system that also serves as an instrument for measuring temperature, humidity and luminosity. The system aims to allow through the IoT (Internet of Things) resource, remotely manage the environments that are under control and monitoring. There are several practical applications for a system that can monitor and, if necessary, activate automated mechanisms to correct these variables in controlled environments, however, it is in home automation that a greater predominance is observed. The device will record in a database the periodic readings carried out over a certain period of time, enabling the creation of tables and graphs. In addition to having an HMI which we can interact with by adjusting a range of parameters for triggering alarms or commands in case of exceeding adjusted units. The prototype was successful in reading the sensors and adjusting the intervals of the sensor units, by means of a serial communication it is able to send the data to an IoT service that stores the data on a web platform that can be accessed by any device with internet access through the ThinkSpeak cloud platform. The activation of the peripherals that will act without control of the environment in a simulation character, will be represented by a set of led's.

Keywords: *Environment control; Temperature; Moisture; Brightness.*

LISTA DE TABELAS

Tabela 1 – Ligação do PIC	36
Tabela 2 – Funções dos botões.....	38
Tabela 3 – Telas do Display	40
Tabela 4 – Bibliotecas auxiliares	42
Tabela 5 – Custos.....	42
Tabela 6 – Grandezas coletadas teste 1	49
Tabela 7 – Grandezas coletadas teste 2	51

LISTA DE FIGURAS

Figura 1 – Intel 4004	22
Figura 2 – Fechadura eletrônica	22
Figura 3 – PIC18F4520	23
Figura 4 – Identificação dos pinos	24
Figura 5 – Módulo Comunicação	24
Figura 6 – Pinos de ligação	25
Figura 7 – Sensor de temperatura e umidade	25
Figura 8 – Pinagem DHT22.....	26
Figura 9 – Sensor de Luminosidade.....	27
Figura 10 – RTC.....	28
Figura 11 – Display de LCD	29
Figura 12 – IDE MikroC.....	30
Figura 13 – Bibliotecas MikroC.....	30
Figura 14 – Proteus 8 Profissional	31
Figura 15 – Interface <i>ThingSpeak</i>	32
Figura 16 – IDE Proteus.....	33
Figura 17 – Circuito Elétrico	34
Figura 18 – Simulação	35
Figura 19 – Serial Monitor	35
Figura 20 – Protoboard	37
Figura 21 – Menu Principal.....	37
Figura 22 – Menu Relógio	38
Figura 23 – Menu Temperatura.....	39
Figura 24 – Menu Umidade.....	39
Figura 25 – Menu Luminosidade	40
Figura 26 – Fluxograma menu	41
Figura 27 – DataLogger Testo 174H.....	42
Figura 28 – Luxímetro Digital MLM-1011 – Minipa.....	42
Figura 29 – Arquivo <i>Gerber</i>	44
Figura 30 – Face Top da placa.....	45
Figura 31 – Protótipo montado	46

Figura 32 – Temperatura.....	47
Figura 33 – Umidade.....	48
Figura 34 – Luminosidade.....	48
Figura 35 – Ambiente de testes 1.....	49
Figura 36 – Grandezas de cruzamento teste 1	51
Figura 37 – Grandezas de cruzamento teste 2	53
Figura 38 – Luminância dos fabricantes.....	54
Figura 39 – Cálculo de lux.....	54
Figura 40 – Cálculo eficiência energética.....	55
Figura 41 – Ambiente de testes 2.....	56
Figura 42 – Teste LED	56
Figura 43 – Teste Fluorescente.....	57
Figura 44 – Teste Halógena	57
Figura 45 – Teste Incandescente	58
Figura 46 – Valores medidos lux	58
Figura 47 – Comparativo de dados teóricos.....	59
Figura 48 – Comparativo de dados práticos.....	60
Figura 49 – Comparativo da eficiência	60
Figura 50 – Lâmpada halógena.....	61
Figura 51 – Lâmpada <i>LED</i>	61
Figura 52 – Lâmpada fluorescente	62
Figura 53 – Lâmpada incandescente	62
Figura 54 – Equivalência de potências.....	63

LISTA DE SIGLAS E ABREVIATURAS

LCD	<i>Liquid Crystal Display</i> (Visor de Cristal Líquido)
LED	<i>Light-Emitting Diode</i> (Diodo Emissor de Luz)
PCB	<i>Print Circuit Board</i> (Placa de Circuito Impresso)
PIC	<i>Peripheral Interface Controller</i> (Controlador de Interface periférica)
GND	<i>Ground</i> (Terra)
RTC	<i>Real Time Clock</i> (Relógio de Tempo Real)
SPI	<i>Serial Peripheral Interface</i> (Interface Periférica Serial)
SDA	<i>Serial Data</i> (Dados Seriais)
SCL	<i>Serial Clock</i> (Relógio Serial)
UART	<i>Universal Asynchronous Receiver Transmitter</i> (Transmissor Receptor Assíncrono Universal)
USB	<i>Universal Serial Bus</i> (Barramento Serial Universal)
LDR	<i>Light Dependent Resistor</i> (Resistor Dependente de Luz)
VCC	<i>Voltage Common Collector</i> (Tensão de Coletor Comum)
CPU	<i>Central Processing Unit</i> (Unidade Central de Processamento)
CLP	Controlador Lógico Programável
UCP	Unidade Central de Processamento
IoT	<i>Internet Of Things</i> (Internet das Coisas)
NTC	<i>Negative Temperature Coefficient</i> (Coeficiente de Temperatura Negativo)
SRAM	<i>Static Random Access Memory</i> (Memória de Acesso Aleatório Estático)
VBAT	<i>Voltage Battery</i> (Tensão da Bateria)
IHM	Interface Homem Máquina
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
APIs	<i>Application Programming Interface</i> (Aplicativo de Interface de Programação)
PTH	<i>Pin Through Hole</i> (Pinagem de Furo Passante)

SUMÁRIO

1	INTRODUÇÃO	18
1.1	Objetivo Principal.....	19
1.2	Objetivo Específico	19
1.3	Organização do Trabalho.....	20
2	FUNDAMENTAÇÃO	21
2.1	Surgimento	21
2.2	O CLP e o microcontrolador.	21
2.3	A Domótica.	22
3	COMPONENTES UTILIZADOS	23
3.1	Microcontrolador.....	23
3.2	Módulo Comunicação.	24
3.3	Sensor de temperatura e umidade.....	25
3.4	Sensor de luminosidade.....	26
3.5	Relógio Serial.	27
3.6	Display de LCD.....	28
3.7	Software.....	29
4	DESENVOLVIMENTO DO PROJETO	33
4.1	Desenvolvimento com simulador.....	33
4.2	Desenvolvimento físico.....	36
4.3	Bibliotecas de programação.....	41
4.4	Custos do Projeto.....	42
5	PROTÓTIPO	44
5.1	Protótipo da placa.....	44
5.2	Montagem da placa.....	45
5.3	Testes.....	46
5.4	Envio e tratamento dados.....	47
6	RESULTADOS	49
6.1	Testes de ambientes	49
6.2	Testes Luminotécnicos.....	53
6.3	Limitações encontradas e conclusão parcial.....	64
7	CONCLUSÃO	65
8	PROJETOS FUTUROS	66
9	REFERÊNCIAS BIBLIOGRÁFICAS	67
	ANEXO A: Código Fonte	69
	ANEXO B: Circuitos Elétricos	78

1 INTRODUÇÃO

A aplicação de sensores de temperatura, umidade e luminosidade é largamente observada em diferentes tipos de projetos, como exemplo pode-se citar a indústria farmacêutica que opera sobre controles rígidos de seus processos, desde a fabricação até o transporte e armazenamento, além de muitos outros de igual rigor, como a indústria de alimentos, ambientes laboratoriais, data centers entre outros; contudo é na domótica e no controle de ambientes que se observa uma maior predominância. Segundo pesquisa no Google Acadêmico® há mais de 6000 citações em língua portuguesa sobre o uso destes três sensores. Entretanto, a utilização simultânea destes três dispositivos é prevalente nas áreas de climatologia e agricultura.

Na busca de soluções tecnológicas que possam minimizar os problemas decorrentes do consumo desenfreado de energia, a indústria segue lançando novos equipamentos e dispositivos cada vez mais eficientes. As lâmpadas de LED são um exemplo clássico deste processo e se tornaram item obrigatório em qualquer projeto de luminotécnico.

Em substituição às lâmpadas tradicionais, por apresentarem melhor eficiência energética Procel [1] as lâmpadas de LED podem ser facilmente encontradas no mercado, de vários modelos e de fabricantes distintos, nacionais e importadas. Tal como ocorre com diversos e outros produtos com forte demanda, a comercialização de itens de baixa qualidade preocupa o mercado de lâmpadas. De acordo com a Proteste [2], entidade civil sem fins lucrativos que atua em defesa do consumidor, os resultados dos testes efetuados em lâmpadas de *LED* de diversos fornecedores no Brasil apresentaram durabilidade menor que a informada e baixo fator de potência, impedindo economia maior de energia. Assim, principalmente as lâmpadas provenientes da China representam uma preocupação especial.

Com base neste cenário espera-se desenvolver um dispositivo que possa servir para o controle de ambientes automatizados além de testar a eficiência das lâmpadas encontradas no mercado e, de alguma forma, contribuir para melhorar a sua regulação.

1.1 Objetivo Principal

O projeto propõe o desenvolvimento de um sistema microcontrolado de monitoramento e controle de ambientes, que realiza medição de variáveis temperatura, umidade e luminosidade. Existem várias aplicações práticas para um sistema que possa monitorar e, se necessário, acionar mecanismos automatizados para a correção dessas variáveis em ambientes controlados. O dispositivo também efetuará a coleta de dados por meio de comunicação serial a qual os dados serão tratados.

1.2 Objetivo Específico

Os objetivos específicos deste trabalho são:

- Desenvolver uma metodologia para a obtenção das grandezas a serem medidas.
- Determinar os instrumentos para a obtenção destas grandezas.
- Implementar um sistema microcontrolado para coleta das grandezas e envio dos dados para uma base de tratamento futuro.
- Desenvolver a programação de um *firmware* que atenda a necessidade e gerencie as medições e cálculos.
- Efetuar simulações do circuito estabelecido.
- Desenvolver métodos de envio e tratativas dos dados coletados.
- Montar o circuito em *protoboard* para ajustes finais.

1.3 Organização do Trabalho

- Capítulo 1: Introdução

Esse capítulo introduz algumas áreas a qual o projeto pode ser implementado.

- Capítulo 2: Fundamentação

Esse capítulo resume a história da automação e do microcontrolador.

- Capítulo 3: Materiais Utilizados

Esse capítulo apresenta os principais sensores, módulos e *softwares* utilizados.

- Capítulo 4: Desenvolvimento do Projeto

Esse capítulo explica o desenvolvimento do projeto passo a passo.

- Capítulo 5: Protótipos

Esse capítulo vislumbra a montagem final do protótipo final.

- Capítulo 6: Resultados

Esse capítulo mostra os resultados coletados e tratados que o trabalho propõe.

- Capítulo 7: Projetos Futuros

Esse capítulo abordara possíveis aplicações futuras para o protótipo.

- Capítulo 8: Conclusão

Esse capítulo apresenta a conclusão a partir dos resultados.

- Capítulo 9: Referências Bibliográficas

Esse capítulo enuncia todas as referências utilizadas no projeto.

2 FUNDAMENTAÇÃO

Para o desenvolvimento deste trabalho, vamos abordar a história e mostrar o porquê a automação de processos está ganhando espaço ao longo dos anos, se tornando uma das áreas mais promissoras hoje em dia.

2.1 Surgimento

Na década de 1960 e início da de 1970, as fábricas automobilísticas começaram uma mudança na sua linha de produção que era composta por uma lógica de contatos, que utilizavam painéis com relés eletromecânicos e muita fiação [3]. Essa mudança se deu para sanar problemas como custo operacional elevado desses painéis, inflexibilidade de alterar a lógica desses painéis e um alto custo de desenvolvimento e manutenção.

Com todos esses problemas a vista a *General Motors* desenvolveu o equipamento que teria algumas vantagens:

- Seria facilmente programado;
- Fisicamente menor;
- Ter uma manutenção mais fácil utilizando módulos (tipo *plug-in*)
- Produzir dados para um sistema de coletas;
- Funcionar em ambientes indústrias com confiabilidade;
- Ter um custo competitivo no mercado em relação a tecnologia empregada na época;

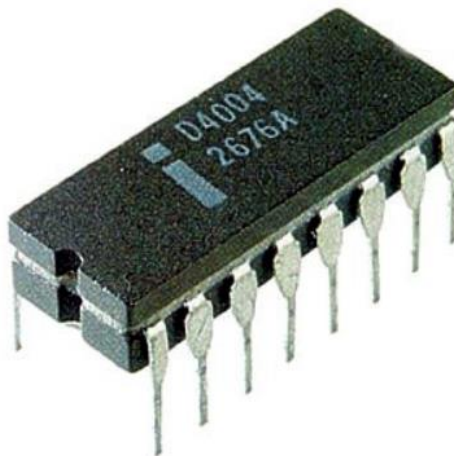
Assim se inventou o primeiro Controlador Lógico Programável (CLP).

2.2 O CLP e o microcontrolador.

O CLP é constituído por uma CPU que é uma unidade central de processamento, que nada mais é que um microprocessador. A Figura 1 mostra

o primeiro microcontrolador comercializado da Intel em 1970.

Figura 1: Intel 4004

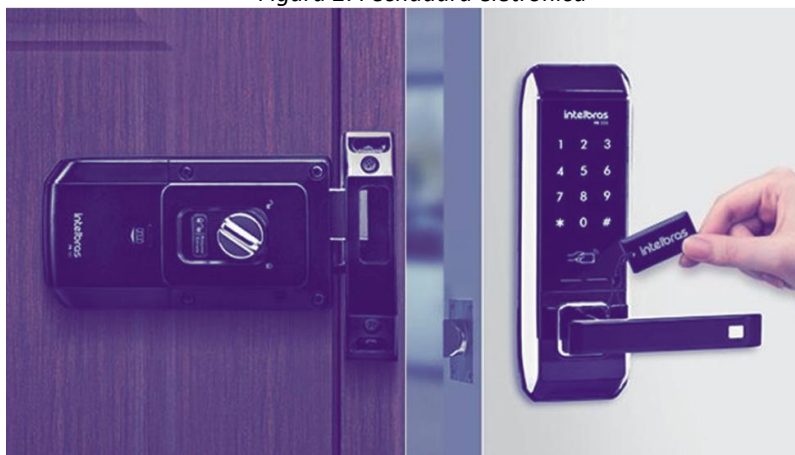


Fonte: https://pt.wikipedia.org/wiki/Intel_4004 acessado 16/03/2021

2.3 A Domótica.

Com o passar dos anos a tecnologia dos microprocessadores foi ficando mais acessíveis em termos financeiros e começou a ser difundida em outros ramos que não são da indústria. Hoje a automação tem uma relação muito próxima com a parte residencial por automatizar ações mecânicas que estão cada dia mais desaparecendo. Um exemplo são as chaves que estão sendo trocadas por acionamentos por senha como mostra a Figura 2.

Figura 2: Fechadura eletrônica



Fonte: <https://www.uol.com.br/tilt/noticias/redacao/2019/06/13/trava-magica-entenda-como-funcionam-as-fechaduras-eletronicas.ht> acessado 16/03/2021

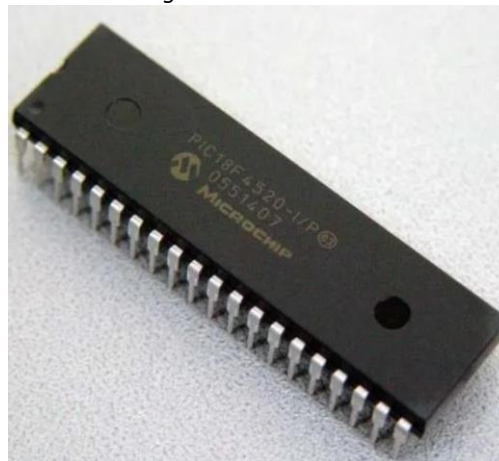
3 COMPONENTES UTILIZADOS

Neste capítulo são descritos os principais componentes que utilizamos para o desenvolvimento deste projeto.

3.1 Microcontrolador.

Para este projeto foi utilizado o microcontrolador PIC18F4520 apresentado na Figura 3, por suas características de memória e processamento que melhor atenderam a necessidades, e o custo. A família PIC18 [4] oferece alto desempenho computacional a um preço econômico com a adição de memória *Flash* aprimorada de alta resistência. A família 4520 apresenta melhorias de design que tornam esses microcontroladores uma escolha lógica para muitos aplicativos sensíveis ao consumo de energia de alto desempenho.

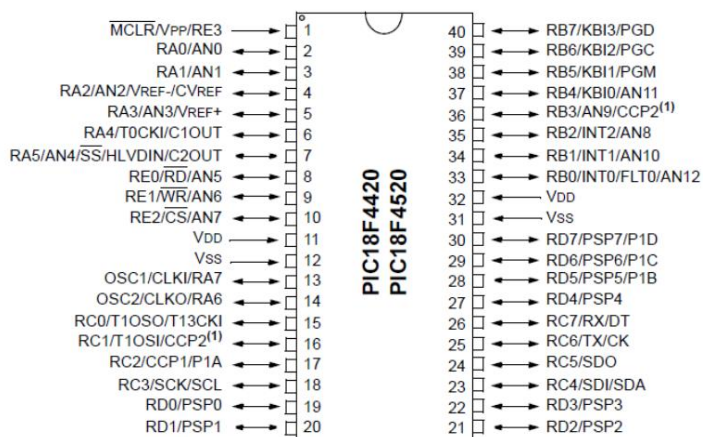
Figura 3: PIC18F4520



Fonte: <https://www.casadarobotica.com/componentes-eletronicos/componentes/microcontrolador/microcontrolador-pic-18f4520-microchip> acessado 16/03/2021

O microcontrolador é composto por vários pinos que podem ser configurados como entradas e saídas, além dos pinos de alimentação e de configurações que serão usados para determinar funções específicas do equipamento apresentado na Figura 4.

Figura 4: Identificação dos pinos

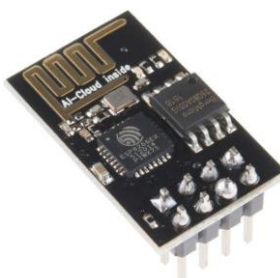


Fonte: <https://www.baudaeletronica.com.br/microcontrolador-pic18f4520-i-p.html> acessado 16/03/2021

3.2 Módulo Comunicação.

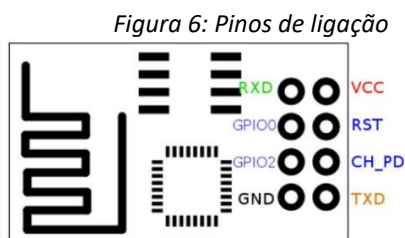
O módulo ESP8266 na Figura 5 é um transceptor sem fio autônomo de baixo custo que pode ser usado para desenvolvimentos de *IoT* de ponto final. Para se comunicar com o módulo ESP8266 [5], o microcontrolador precisa usar um conjunto de comandos AT. O microcontrolador se comunica com o módulo ESP8266-01 usando UART com taxa *Baud* especificada. O módulo ESP8266 funciona apenas com 3,3 V, qualquer coisa acima de 3,7 V danificaria definitivamente o módulo, exigindo, por isto, grande cuidado no momento do uso e utilizando, por exemplo, um circuito divisor de tensão. Portanto, é importante fazer um pequeno regulador de tensão para 3,31 V que possa fornecer no mínimo 500mA. Utilizamos em nosso circuito o LM317, que pode realizar o trabalho facilmente.

Figura 5: ESP8682



Fonte: <https://blog.eletrogate.com/iot-com-modulo-wifi-esp8266-basico/> acessado 16/03/2021

Os pinos de ligação estão apresentados na Figura 6:

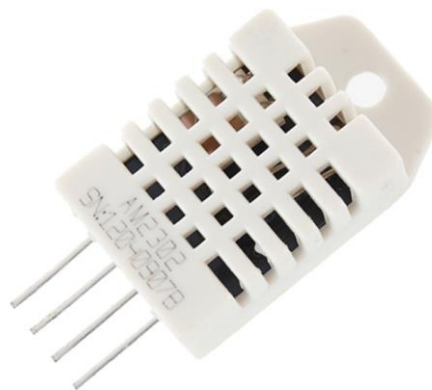


Fonte: <https://blog.eletrogate.com/iot-com-modulo-wifi-esp8266-basico/> acessado 16/03/2021

3.3 Sensor de temperatura e umidade.

Foi utilizado para leitura de temperatura e umidade o sensor DHT22 apresentado na Figura 7. É um dispositivo de baixo custo usado para medição de umidade e temperatura do ar. O sensor de umidade é capacitivo e o sensor de temperatura é um termistor NTC, isto é um resistor sensível a variações de temperatura. Dentro do sensor existe um microcontrolador que faz as medições e transmite os valores no formato digital através de um pino de saída.

Figura 7: Sensor de temperatura e umidade



Fonte: <https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/> acessado 16/03/2021

Especificações do sensor DHT22 [6]:

Faixa de umidade relativa: de 0 a 100 %

Precisão na umidade: ± 2 % UR

RH Faixa de temperatura: -40 a 80 °C

Precisão na temperatura: $\pm 0,5 \%$ °C

Resolução: 0,1

Tempo de resposta: 2 segundos

Alimentação: de 3,5 V a 5 V

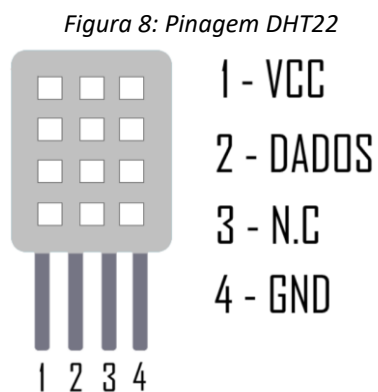
Consumo máximo de corrente: 2,5 mA VCC = 3,5 a 5V

DATA = comunicação de dados

NC = sem conexão

GND = terra

Os pinos de ligação estão apresentados na Figura 8:



Fonte: <https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/>
acessado 16/03/2021

3.4 Sensor de luminosidade.

O Sensor de Luminosidade na Figura 9 se trata de um resistor variável cuja sua resistência varia de acordo com a intensidade da luz. Quanto mais luz incidir sobre o componente, menos a resistência. Possui 2 terminais e não tem polaridade para sua ligação. Em sua composição existe um semicondutor de alta resistência [7].

Figura 9: Sensor de Luminosidade



Fonte: <https://www.microwat.com.br/350> acessado 16/03/2021

O seu funcionamento consiste na incidência de luz em um material semicondutor fazendo com que os fótons desloquem os elétrons desse material criando uma corrente elétrica e assim fazendo a resistência diminuir.

A mudança de estado do *LDR* não é instantânea, o dispositivo possui um tempo de variação, esse tempo chega a 10ms para chegar a baixa resistência.

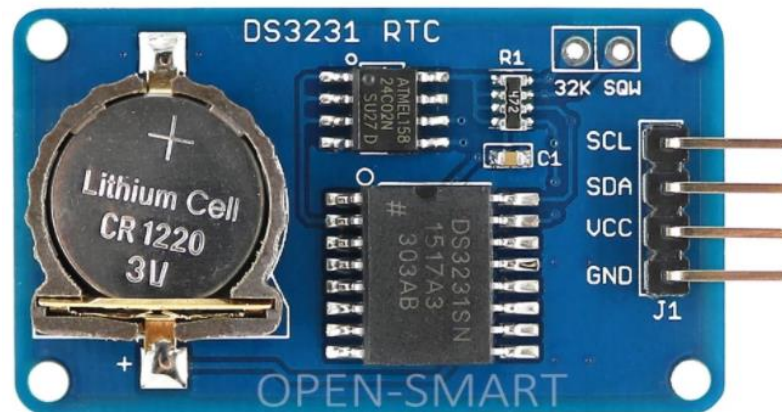
Existem 2 tipos de LDR, um para luz ultravioleta e outro para a luz infravioleta. Sua curva característica é linear, conforme a incidência de luz sobre o sensor, suas características resistivas alteram variando a tensão em sua saída, sendo possível medir as características luminosas do ambiente. Como nos semicondutores também existem 2 tipos de fotocondutores, intrínsecos utilizando materiais sem impurezas e o extrínseco que são semicondutores dopados [8].

3.5 Relógio Serial.

O DS1307 [9] é um relógio / calendário de baixa energia com 56 bytes de *SRAM* com bateria apresentado na Figura 10. O relógio / calendário fornece informações de segundos, minutos, horas, dia, data, mês e ano. O DS1307 opera como um dispositivo escravo no I2C. O acesso é obtido implementando uma condição *START* e fornecendo uma identificação do dispositivo seguido por um endereço de registro. Os registros subsequentes podem ser acessados sequencialmente até que uma condição *STOP* seja executado. Quando o VCC fica 25% abaixo da tensão da bateria, o dispositivo encerra um acesso em

andamento e reinicia o dispositivo contador de endereços. As entradas para o dispositivo não serão reconhecidas neste momento para evitar que dados errados sejam gravados no dispositivo a partir de um sistema fora da tolerância. Quando VCC cai abaixo de VBAT, o dispositivo muda para um modo de bateria reserva de baixa corrente. Ao ser ligado, o dispositivo muda de bateria para VCC quando VCC é maior que VBAT + 0,2 V e reconhece entradas quando VCC é maior que 25% da tensão de bateria.

Figura 10: RTC



Fonte: <https://pt.aliexpress.com/item/32669670601.html> acessado 16/03/2021

3.6 Display de LCD.

O Display LCD 20x4 da Figura 11 é utilizado em projetos onde se necessita uma interface homem-máquina (IHM). Possui 4 linhas com 20 colunas com escrita na cor branca, e sua *backlight* (luz de fundo) azul. O mesmo possui um controlador que pode realizar as operações com 4 ou 8 bits paralelamente. Possui suporte para comunicação I2C. Seus pinos se dividem em entradas/saídas, alimentação, ajustes de contraste, ao todo possui 16 pinos. Suas principais utilizações são para saída de dados.

Principais Características [10]:

Tensão de trabalho: 4,5V ~ 5,5V;

Corrente de trabalho: 1,0mA ~ 1,5mA (*backlight* desligado);

Corrente do *backlight*: 75mA ~ 200mA;

Controlador: HD447780 *Backlight* AZUL e escrita BRANCA;

Dimensões: 98 x 60 x 14 mm (C x L x A);

Área do visor: 76 x 26 mm;

Tamanho do ponto: 0,54 x 0,54 mm;

Tamanho do caractere: 2,94 x 4,74 mm.

Figura 11: Display de LCD



Fonte: <https://www.microwat.com.br/713> acessado 16/03/2021

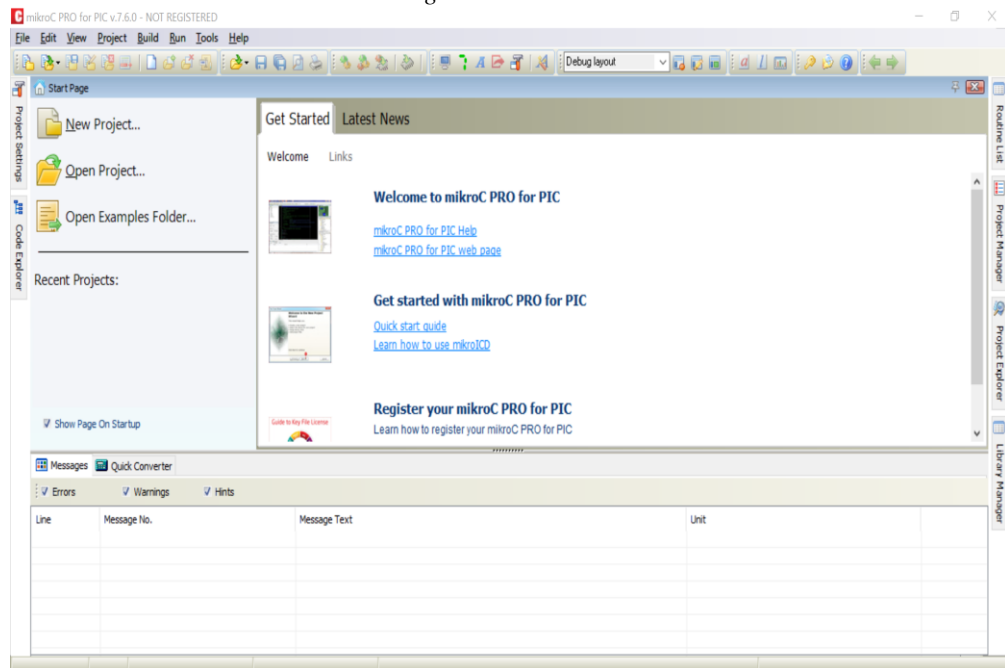
3.7 Software.

Os principais *softwares* utilizados nesse projeto para a programação e a montagem do circuito foram o MikroC, Proteus e o Excel. Esses *softwares* são descritos a seguir.

3.7.1 MikroC.

A Figura 12 mostra o compilador MikroC [11] que é uma IDE de desenvolvimento de códigos fontes que foi utilizado para a construção da programação do PIC. Através desse *software* a programação é compilada e carregada para o PIC executar as funções e controlar os sensores e outros dispositivos. Utilizamos esse *software* devido a familiaridade que tivemos nas aulas de microcontrolador. O código desenvolvido por ele se encontra no Anexo A.

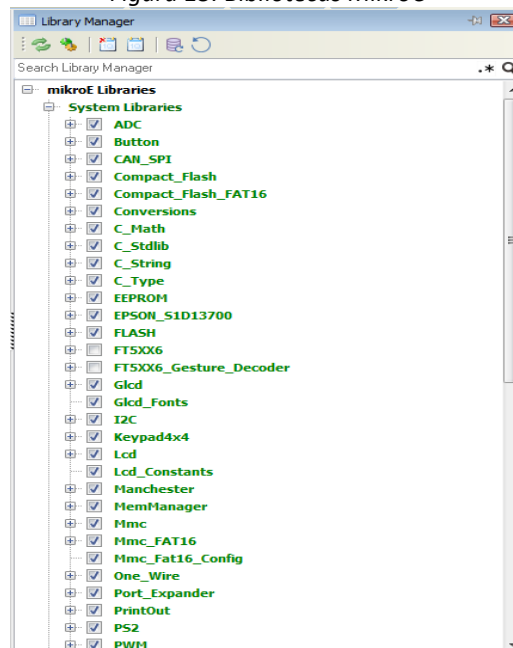
Figura 12: IDE MikroC



Fonte: Autores

O software é de simples manejo e a própria IDE traz bibliotecas com as funções mais utilizados para simplificar a programação, como mostra a Figura 13.

Figura 13: Bibliotecas MikroC



Fonte: Autores

3.7.2 Proteus.

O *software* Proteus [12] foi utilizado para simular o circuito virtualmente e com essa ferramenta também é possível realizar simulações reais. A Figura 14 mostra o logo tipo do programa.

Figura 14: Proteus 8 Professional



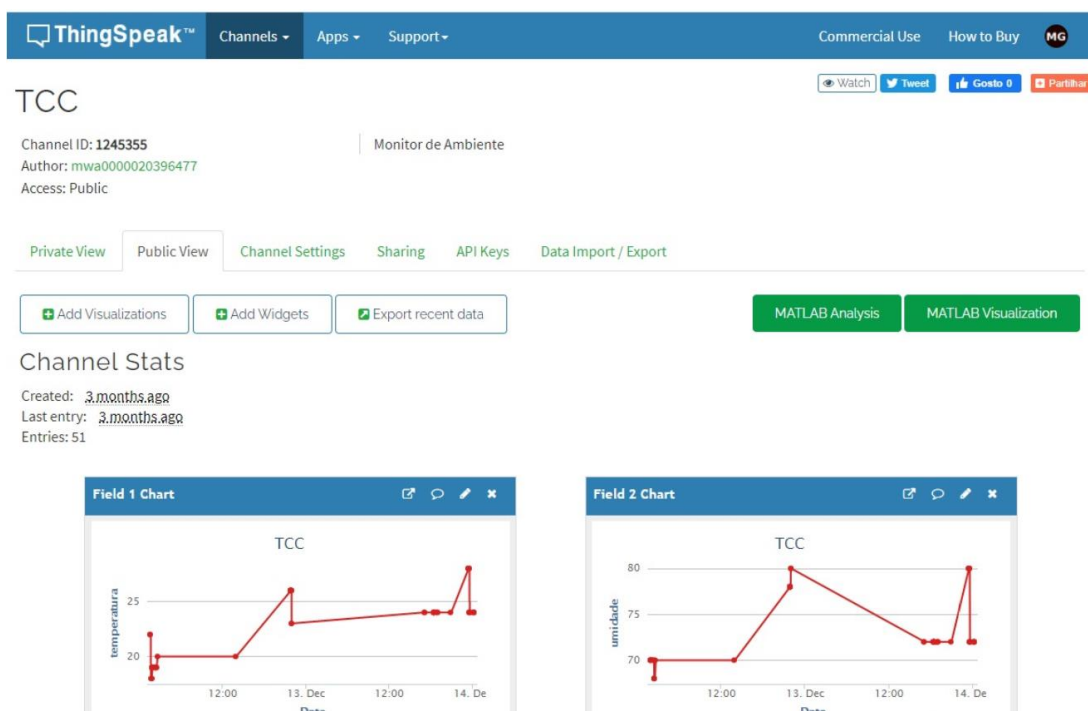
Fonte: <https://www.labcenter.com/> acessado 25/05/2021

3.7.3 ThingSpeak.

ThingSpeak é uma página que funciona como um banco de dados, que recebe informações geradas por dispositivos *IoT* e as disponibiliza para acesso em forma de gráfico podendo ser acessado por qualquer navegador *web*. Isto torna muito mais fácil o armazenamento, obtenção e visualização da informação, principalmente nos casos de IP (Protocolo de Internet) flutuante e redes locais roteadas através de um único IP, o dispositivo que gera a informação não precisa ser acessado diretamente.

Com o *ThingSpeak* podemos visualizar os dados em forma de gráficos e vários outros formatos e APIs. O elemento básico é um canal onde os dados são armazenados tratados e apresentados. A imagem da Figura 15 mostra a interface da página *web*.

Figura 15: Interface ThingSpeak



Fonte: <https://thingspeak.com/channels/1245355> acessado 16/03/2021

4 DESENVOLVIMENTO DO PROJETO

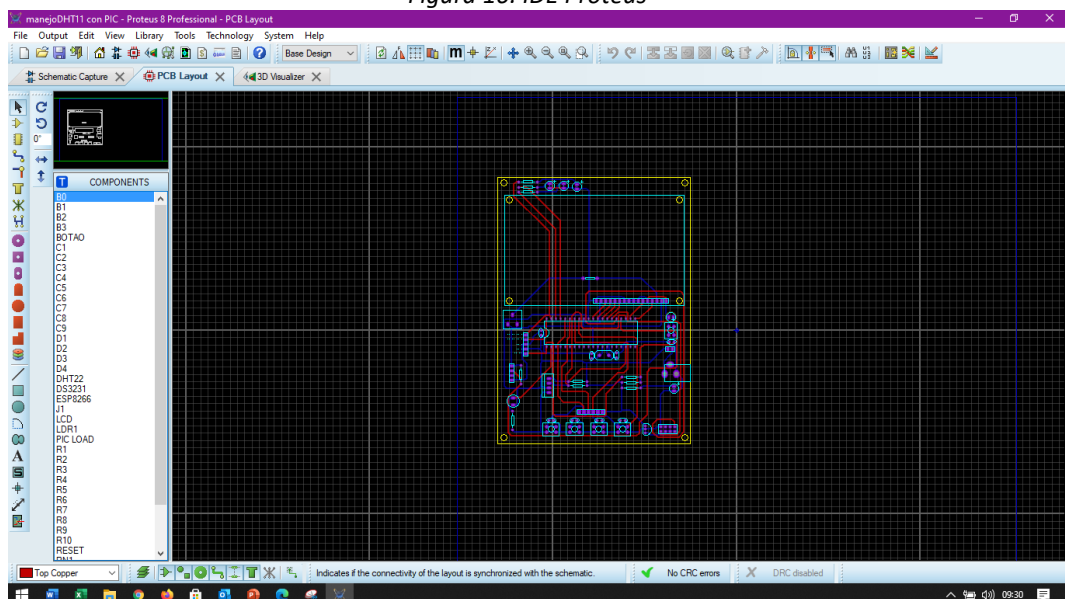
Este capítulo inclui o funcionamento do dispositivo explicado em três frentes de desenvolvimento, que são o desenvolvimento em *software* de simulação com testes, o desenvolvimento físico em *protoboard*, para testes dos componentes e *firmware* em condições reais e o protótipo final desenvolvido em placa de circuito impresso será abordado no próximo capítulo devido a relevância do mesmo.

4.1 Desenvolvimento com simulador.

Inicialmente foi realizada a simulação do circuito com o *software* de simulação Proteus, onde foram realizados todos os testes dos componentes a serem utilizados, e simulações durante o processo de programação do *firmware*, em um ambiente virtual.

A Figura 16 mostra a área de trabalho inicial onde foi realizado a montagem do circuito para testes, selecionando os componentes e definindo a área de montagem.

Figura 16: IDE Proteus

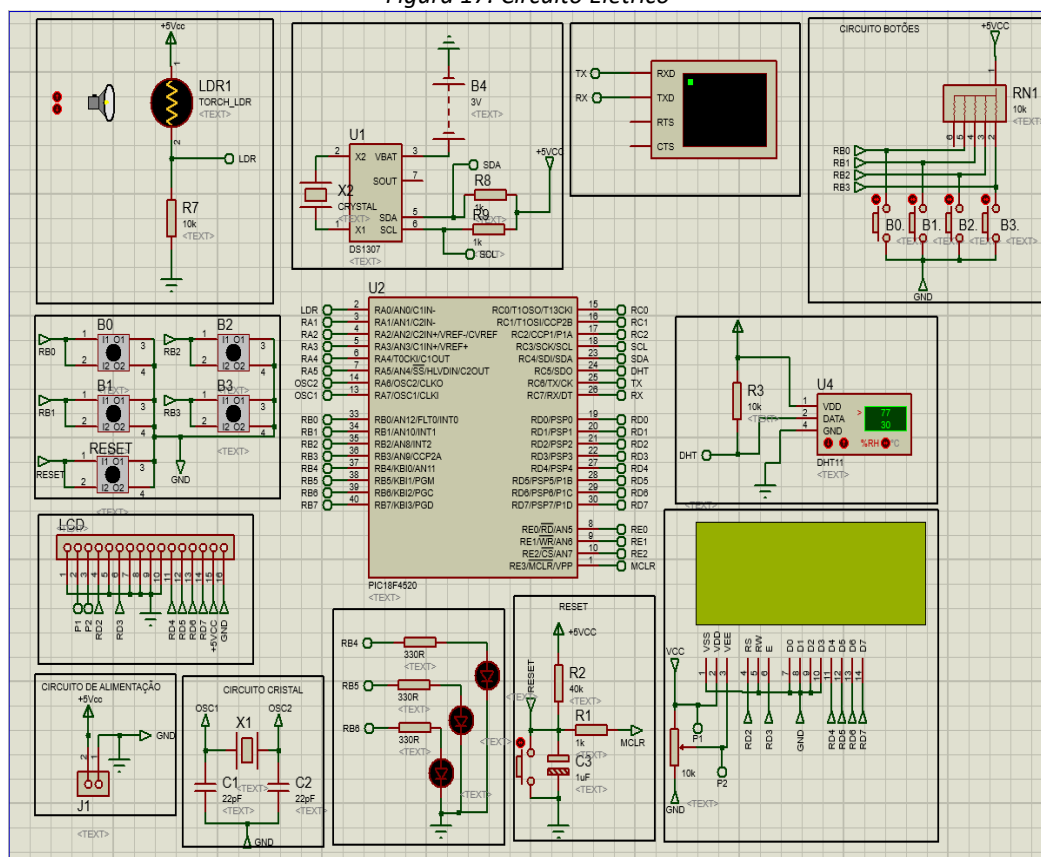


Fonte: Autores

Figura 17 apresenta os circuitos montados para realizar os testes virtuais para a verificação do funcionamento do código fonte, estes circuitos foram montados em módulos separados para uma melhor visualização e entendimento, com uma descrição detalhada dos pinos do microcontrolador e seus respectivos componentes, separados por circuitos individuais. Qualquer alteração ou atualização não afeta o restante do projeto.

Cada bloco está separado e apresentado individualmente no Anexo B.

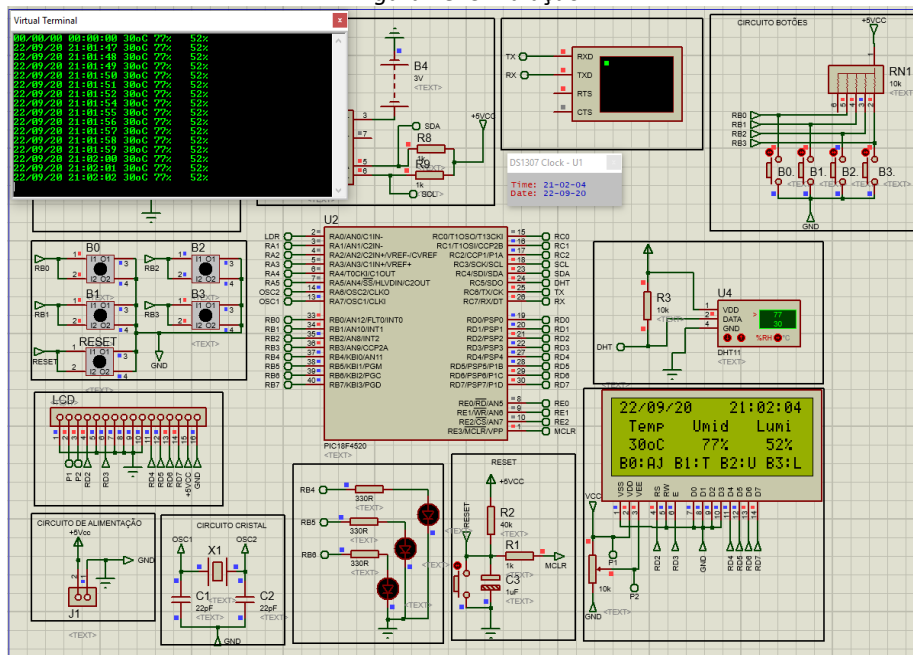
Figura 17: Circuito Elétrico



Fonte: Autores

A Figura 18 traz uma simulação feita no *software* de simulação Proteus, mostrando no *display*, os valores medidos pelos sensores, a contagem de tempo efetuada pelo módulo RTC, o funcionamento dos *LEDS* e botões, e o envio dos dados coletados através do *virtual terminal* através da interface UART, sem a necessidade de conexão física, fios ou conectores, o que possibilita uma maior flexibilidade e confiabilidade para a montagem do circuito, sem os riscos que a montagem física oferece, como curto circuitos ou queima de componentes ou circuitos abertos.

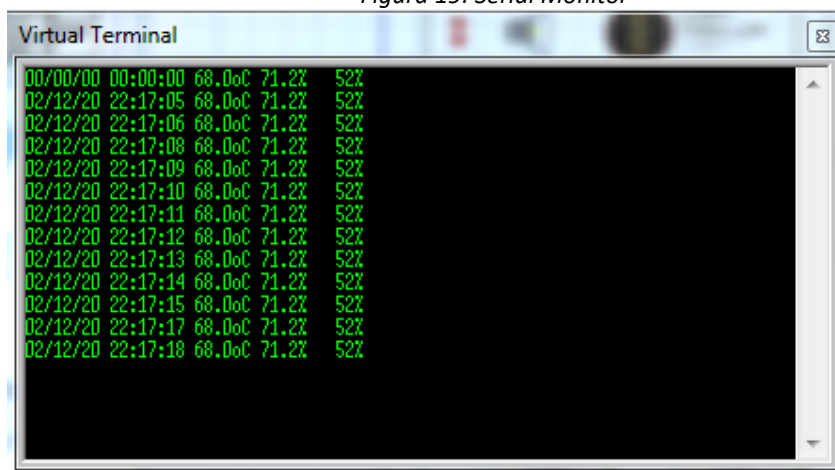
Figura 18: Simulação



Fonte: Autores

A Figura 19 apresenta o envio de dados pela comunicação serial apresentados no *virtual terminal* do software de simulação Proteus, que indica que os dados estão sendo enviados através da interface RX/TX, configurados nos pinos RX25 e TX26 no microcontrolador. Os dados enviados são a data, hora, temperatura, umidade e luminosidade.

Figura 19: Serial Monitor



Fonte: Autores

A montagem dos pinos de entrada/saída do microcontrolador PIC está detalhada na Tabela 1.

Tabela 1: Ligação PIC

PIC 18F4520	
Circuito	pinos
LDR	RA0
Botões	RB0 ao RB3
Oscilador	RA6 e RA7
Display de LCD	RD2 ao RD7
Módulo DHT 22	RC5
Módulo DS1307	RC3 e RC4
Módulo ESP8622	RC6 e RC7

Fonte: Autores

4.2 Desenvolvimento físico.

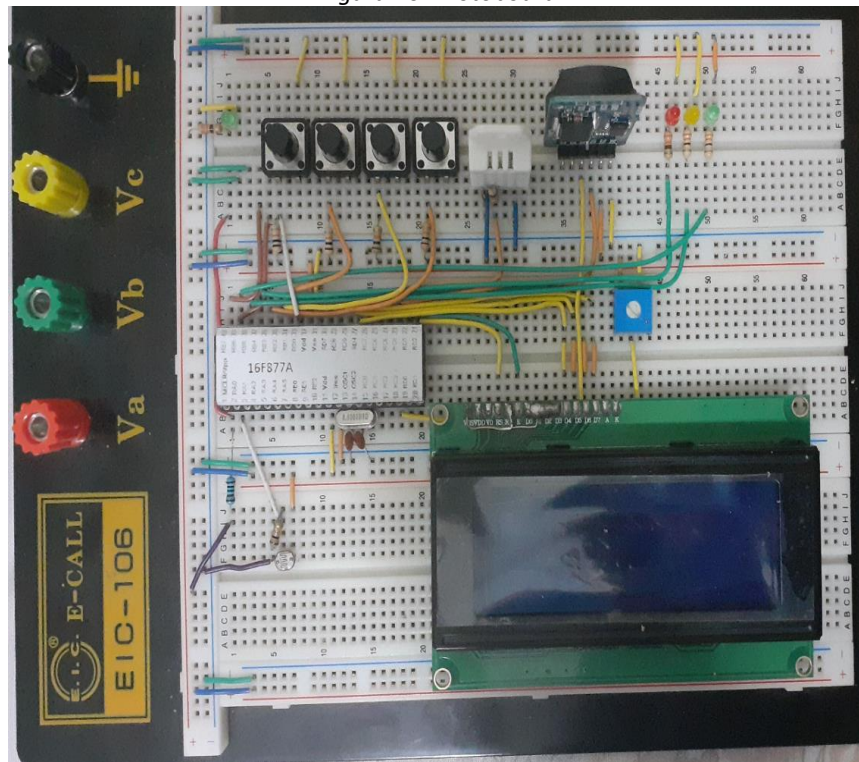
A segunda etapa é a montagem em *protoboard* como mostra a Figura 20, com todos os componentes que citamos na parte teórica desse trabalho, para que sejam realizados testes de funcionamento com componentes reais em condições de uso com todas as variações elétricas e físicas que os componentes sofrem, como *stress*, aquecimento e consumo de energia.

Estes componentes foram escolhidos, seguindo critérios recomendados pelos fabricantes dos dispositivos, porém foram necessárias algumas alterações para compensar perdas por mau contato, ou falha no processo de fabricação de determinado componente.

A fase de montagem em *protoboard* (Protótipo), facilita a troca de componente caso não seja o ideal para a aplicação.

Neste caso específico, deve-se tomar um cuidado com a organização dos cabos de ligação, para facilitar a identificação em caso de medições elétricas e de testes.

Figura 20: Protoboard



Fonte: Autores

Quando o equipamento é energizado a primeira imagem que aparece no *display* é uma saudação *HELLO WORD*, todos os *leds* acendem como forma de teste, durante cinco segundos será a tela apresentada, em seguida o dispositivo apresenta a tela menu principal, que contém a data, a hora e as leituras de temperatura, umidade e luminosidade, e a instrução do Botão B3:Menu que é um botão de navegação para acessar os menus de configuração retratado na Figura 21 porém é uma tela que não é configurável, somente de saída dos dados lidos em tempo real.

Figura 21: Menu Principal



Fonte: Autores

Pressionando o Botão B3 uma vez, o dispositivo apresenta a primeira tela de configuração, Tela Ajusta Relógio, mostrada na Figura 22, que apresenta um menu para ajuste do relógio, caso a data e a hora estejam desconfiguradas.

O botão B0 seleciona a grandeza que será alterada, e o botão B1 para dezenas mais e B2 para unidades e B3 sair.

Figura 22: Menu Relógio



Fonte: Autores

A Tabela 2 demonstra os botões para ajuste do relógio e suas respectivas funções.

Tabela 2: Funções dos botões

Função dos botões	
B0	Seleciona
B1	Incrementa dezenas
B2	Incrementa unidades
B3	Próximo menu

Fonte: Autores

Pressionando novamente o botão B3, o dispositivo apresenta a próxima tela de menu Temperatura, na Figura 23, onde é apresentado o valor da temperatura lida pelo sensor DHT e o intervalo de tolerância mínimo e máximo.

Quando estes intervalos de máximo e mínimo são atingidos um led acende dando uma interpretação visual, apontando que os intervalos de máximo ou mínimo foram superados.

A configuração deste intervalo é definida pelo botão B1 que incrementa os valores de mínimo indo de 0 a 99. E o botão B2 incrementa os valores de máximo indo de 0 a 99.

Figura 23: Menu Temperatura



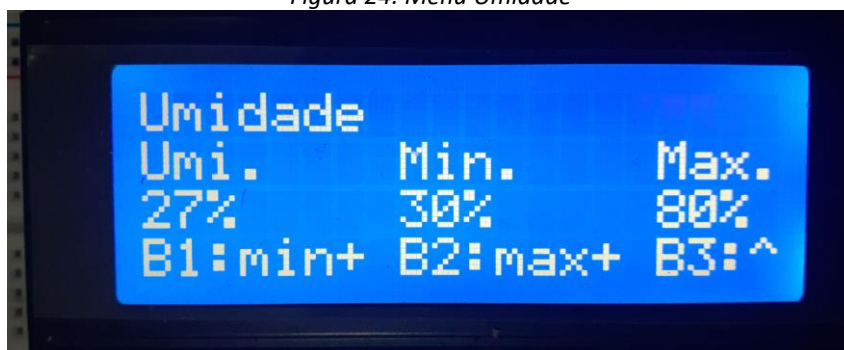
Fonte: Autores

Pressionando novamente o botão B3, o dispositivo apresenta a próxima tela de menu Umidade, na Figura 24, onde é apresentado o valor da umidade lida pelo sensor DHT e o intervalo de tolerância mínimo e máximo.

Quando estes intervalos são superados o *led* amarelo acende dando uma interpretação visual, apontando que os intervalos de máximo ou mínimo foram superados.

A configuração deste intervalo é definida pelo botão B1 que incrementa os valores de mínimo indo de 0 a 99. E o botão B2 incrementa os valores de máximo indo de 0 a 99.

Figura 24: Menu Umidade



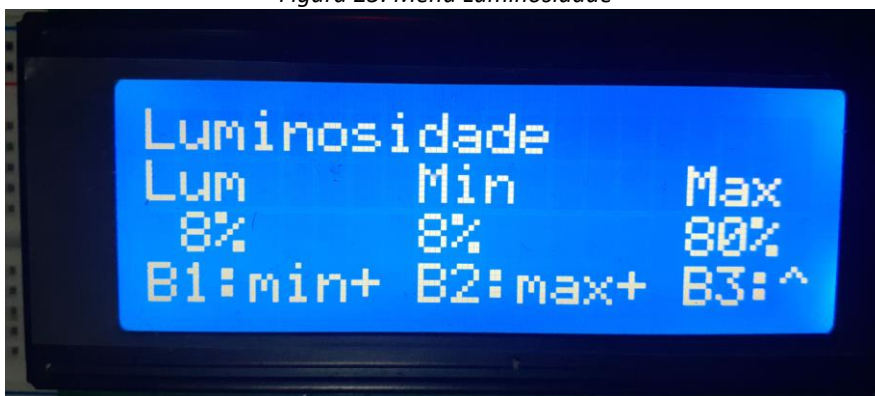
Fonte: Autores

Pressionando novamente o botão B3, o dispositivo apresenta a próxima tela de menu Luminosidade, na Figura 25, onde é apresentado o valor da luminosidade lida pelo sensor LDR e o intervalo de tolerância mínimo e máximo.

Quando estes intervalos são superados o led verde acende dando uma interpretação visual, apontando que os intervalos de máximo ou mínimo foram superados.

A configuração deste intervalo é definida pelo botão B1 que incrementa os valores de mínimo indo de 0 a 99. E o botão B2 incrementa os valores de máximo indo de 0 a 99.

Figura 25: Menu Luminosidade



Fonte: Autores

A Tabela 3 mostra todas as Telas que compõem o projeto que serão visualizados no *Display* LCD.

Tabela 3: Telas do Display

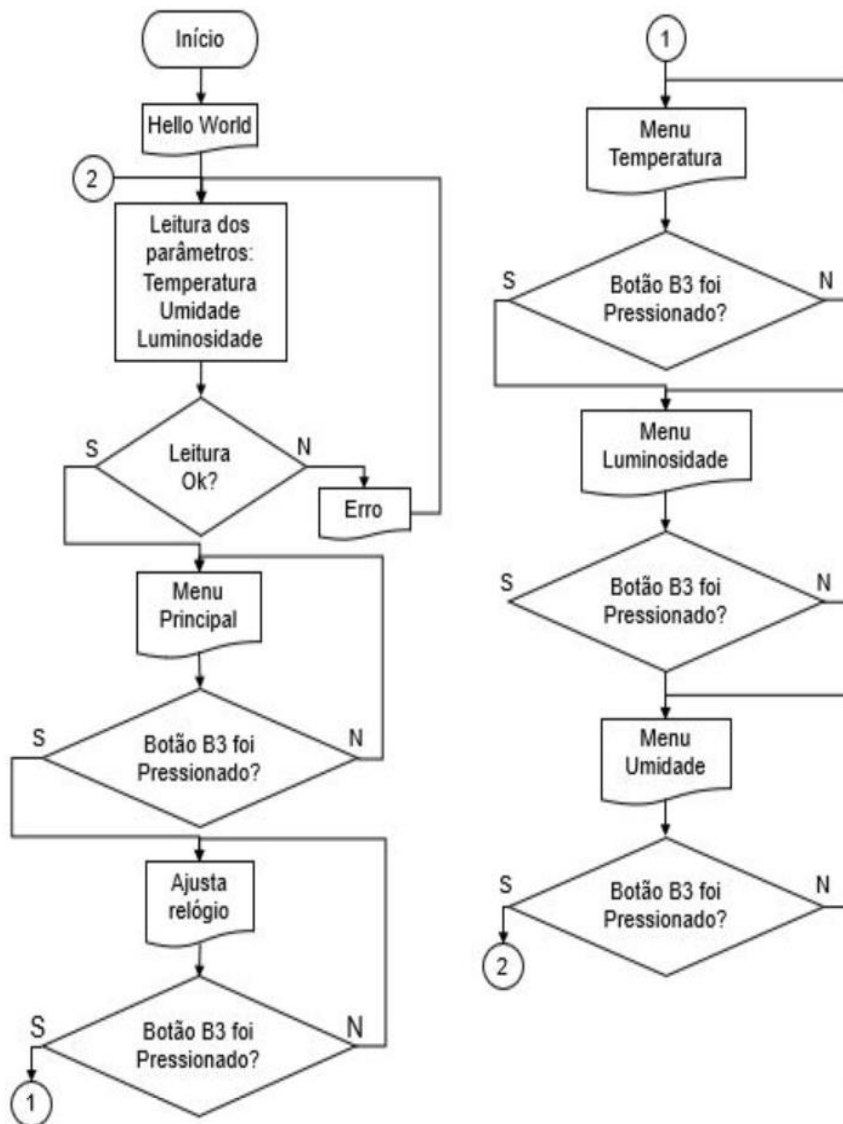
Telas	
1	Menu Principal
2	Ajuste Relógio
3	Ajuste Temperatura e Umidade
4	Ajuste Luminosidade

Fonte: Autores

Para um melhor entendimento da dinâmica e do funcionamento do sistema implementado foi elaborado um fluxograma que demonstra as etapas de funcionamento descritas na Tabela 3 das telas de inicialização e o fluxo de

leituras e menus, conforme a Figura 26.

Figura 26: Fluxograma menu



Fonte: Autores

4.3 Bibliotecas de programação.

Algumas bibliotecas foram utilizadas para a programação do projeto. A maioria delas foi adquirida no próprio compilador que possui uma variedade de bibliotecas prontas para o seu uso, conforme a Tabela 4 que vai descrever as bibliotecas que usamos.

Tabela 4 – Bibliotecas auxiliares

Dispositivos	Bibliotecas
DS3231	I2C
Display LCD	LCD
LDR	ADC
ESP8266	UART

Fonte: Autores

4.4 Custos do Projeto.

Dispositivos disponíveis no Mercado com as mesmas características são caros, e não possuem todas as funcionalidades, em uma pesquisa foram encontrados inúmeros dispositivos mas que não coletam todos os dados, como os das Figuras 27 e 28.

Figura 27: DataLogger NTC -20 a 70°C Umidade 0 a 100%rF* Testo 174H



Fonte: <https://www.testo.com/> acessado 25/05/2021

Figura 28: Luxímetro Digital MLM-1011 – Minipa.



Fonte: <https://www.minipa.com.br/temperatura-e-ambiente/luxímetros/138-mlm-1011> acessado 25/05/2021

O dispositivo da Testo (Figura 27) tem um valor de mercado de R\$ 1099,00, já o dispositivo da Minipa (Figura 28) tem um valor aproximado de R\$273,00.

Para realizar a montagem do projeto foi utilizado todos os dispositivos elétricos que vimos nos capítulos anteriores e aqui vamos mostrar os valores estimados dos custos conforme a Tabela 5.

Tabela 5: Custos

Unidades	Componentes	Valores	Local
1	MICROCONTROLADOR PIC18F4520	R\$36,00	Baú da Eletrônica
1	SENSOR DE LUMINOSIDADE	R\$1,50	Filipeflop
1	SENSOR DHT22 DE TEMPERATURA E UMIDADE	R\$16,90	Mult Comercial
1	RTC 1307	R\$23,90	Filipeflop
4	BOTÕES	R\$1,60	Filipeflop
1	TRIMPOT 10K	R\$2,10	Filipeflop
1	MÓDULO COMUNICAÇÃO ESP8682	R\$22,27	Mult Comercial
20	RESISTORES, CAPACITORES, CRYSTAL E LED	R\$8,00	Mult Comercial
1	DISPLAY LCD 20X4	R\$39,90	Filipeflop
1	PLACA DE CIRCUITO IMPRESSO	R\$76,00	JLCPCB
TOTAL		R\$228,97	

Fonte: Autores

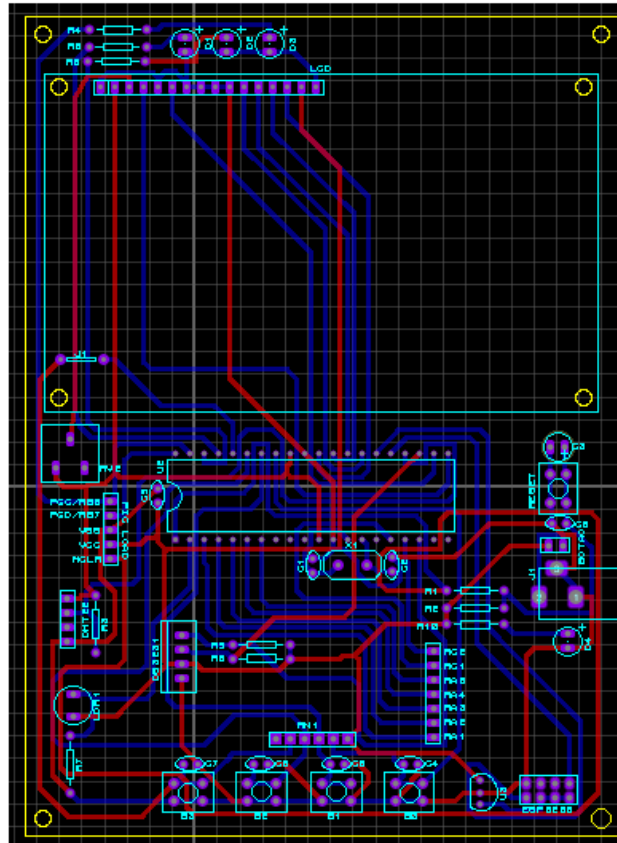
5 PROTÓTIPO

Este capítulo apresenta o protótipo final, testes e a análise dos dados obtidos pela leitura dos sensores.

5.1 Protótipo da placa.

Iniciou-se a realização dos testes em *software* de desenvolvimento, e a obtenção de dados satisfatórios, foi desenvolvido o circuito em *protoboard*, para testes do circuito e componentes em situação real de funcionamento, após atingir os resultados satisfatórios iniciou o desenvolvimento do protótipo do dispositivo partindo do arquivo *Gerber*, conforme a Figura 29 gerado através do *software* simulador Proteus.

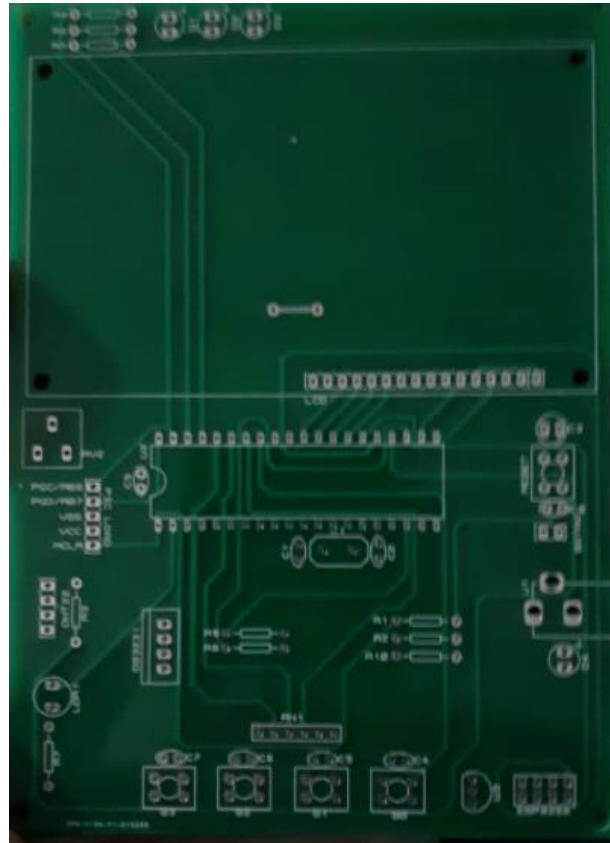
Figura 29: Arquivo Gerber



Fonte: Autores

Devido a complexidade da placa e falta de recursos para sua manufatura, foi enviado o arquivo para desenvolvimento na China chegando ao resultado de uma placa dupla face com furos metalizados conforme a Figura 30.

Figura 30: Face Top da placa



Fonte: Autores

5.2 Montagem da placa.

Com a montagem iniciada a fixação dos componentes manualmente um a um com a tecnologia *PTH*, posicionando manualmente e fixando com solda de estanho e chumbo, chegando ao resultado apresentado na Figura 31.

Figura 31: Protótipo montado



Fonte: Autores

5.3 Testes.

Nos testes efetuados o dispositivo desenvolvido apresentou as grandezas lidas pelos sensores, o dispositivo ficou ligado entre 00:00 horas do até as 23:00, efetuando a coleta dos dados e envio das informações de maneira autônoma satisfatória.

As grandezas apresentadas em tela, foram enviadas via UART, através de comandos AT, para um módulo ESP8266, que direciona os dados para a página da *web* <https://thingspeak.com>, que trata os dados gerando gráficos através da ferramenta *matlab*.

Esse padrão possibilita a criação de gráficos de variáveis onde é possível analisar o comportamento do ambiente em um intervalo determinado de medições.

Tais métricas possibilitam compreender também como cada grandeza se comporta uma em relação a outra em cada etapa do dia.

5.4 Envio e tratamento dados.

A partir dos dados coletados, através do sistema *IoT*, Internet das coisas, com o auxílio de um módulo ESP-01, através de comandos AT, enviados via *UART*, pelo Microcontrolador, enviamos os dados para o site *ThingSpeak*, que nos permite acompanhar os dados de forma remota.

Os comandos utilizados foram:

```
AT+CWMODE=1
```

```
AT+CWJAP="Brunaemilton","21042011"
```

```
AT+CIPMUX=1
```

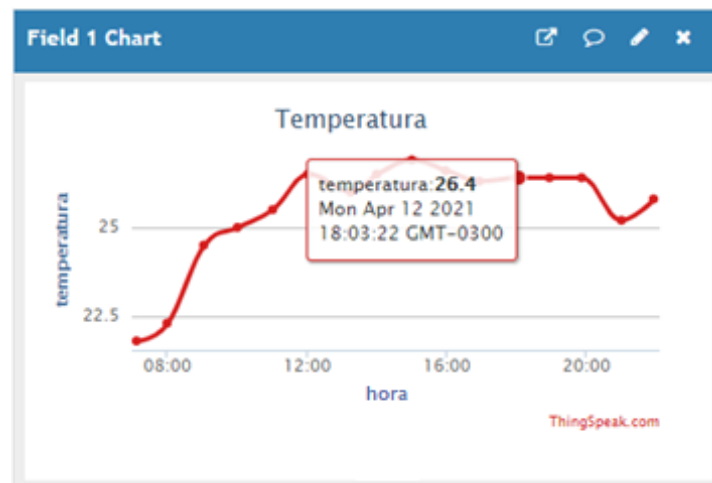
```
AT+CIPSTART=4,"TCP","184.106.153.149",80
```

```
AT+CIPSEND=4,64
```

```
GET/update?key=U7O9ES70E4JEK9DD&field1=24&field2=72&field3=28
```

O primeiro gráfico da Figura 32 mostra a temperatura em função do tempo.

Figura 32: Temperatura



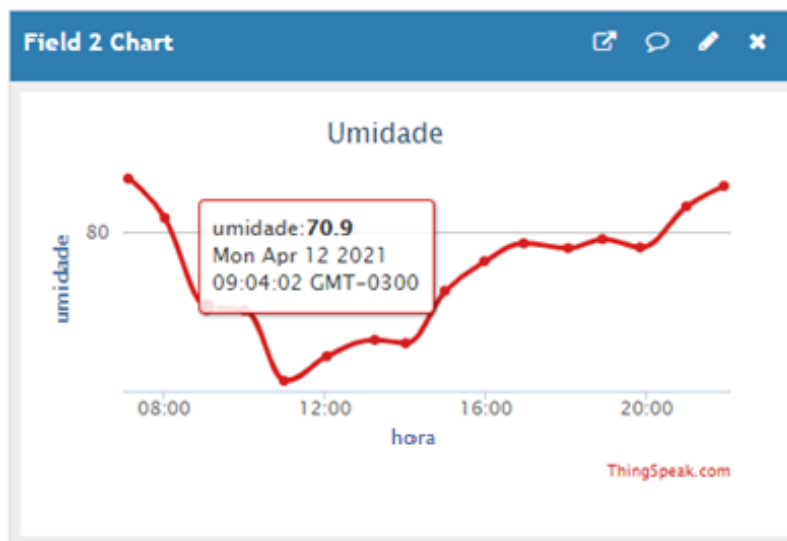
Fonte: Autores

O aumento da temperatura está relacionado diretamente ao aumento da umidade representado na Figura 33 e o ambiente onde o dispositivo está alocado.

Com bases nestes dados, é possível traçar uma média de alteração de

temperatura e umidade em diferentes etapas do dia, otimizando a utilização de recursos como ar condicionado e umidificador.

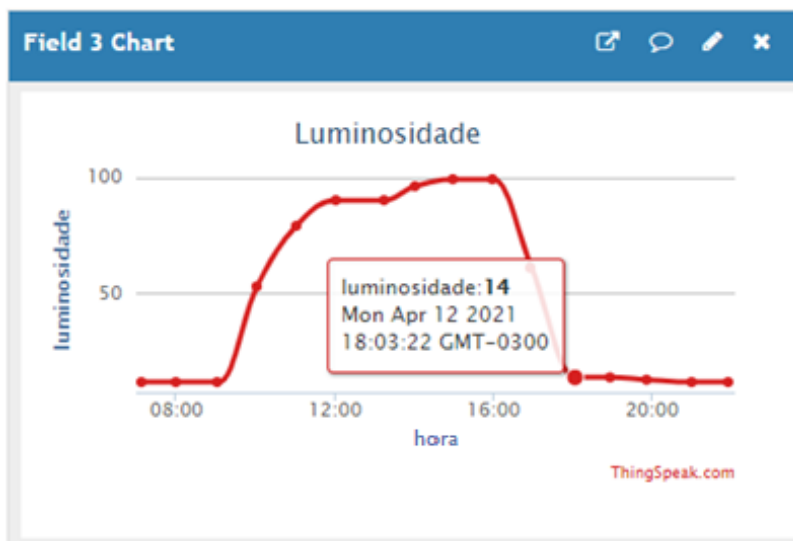
Figura 33: Umidade



Fonte: Autores

Relacionando a terceira medida referente a luminosidade podemos constatar as alterações de luminosidade que encontra no decorrer do dia segundo a Figura 34.

Figura 34: Luminosidade



Fonte: Autores

Através destas medidas, é possível adotar procedimentos para maior eficiência luminosa de um determinado ambiente, e se existe a necessidade de luz artificial ou não.

6 RESULTADOS

Este capítulo demonstra os dados obtidos em testes efetuados em diversos ambientes e condições para várias finalidades.

6.1 Testes de ambientes

Durante os testes iniciais, o dispositivo foi colocado em um ambiente interno com iluminação natural, janela de vidro fechada, mostrado na Figura 35, com um medio controle do ambiente onde ele ficou ligado por 24 horas gerando dados sobre este ambiente e sua variação no decorrer do dia.

Figura 35: Ambiente de teste 1



Fonte: Autores

Após este período, os dados coletados foram baixados do banco de dados *Thingspeak*, em um arquivo no formato *.CSV*, e tratados através do *software excel* no formato de tabelas conforme a Tabela 6.

Tabela 6: Grandezas coletadas teste 1

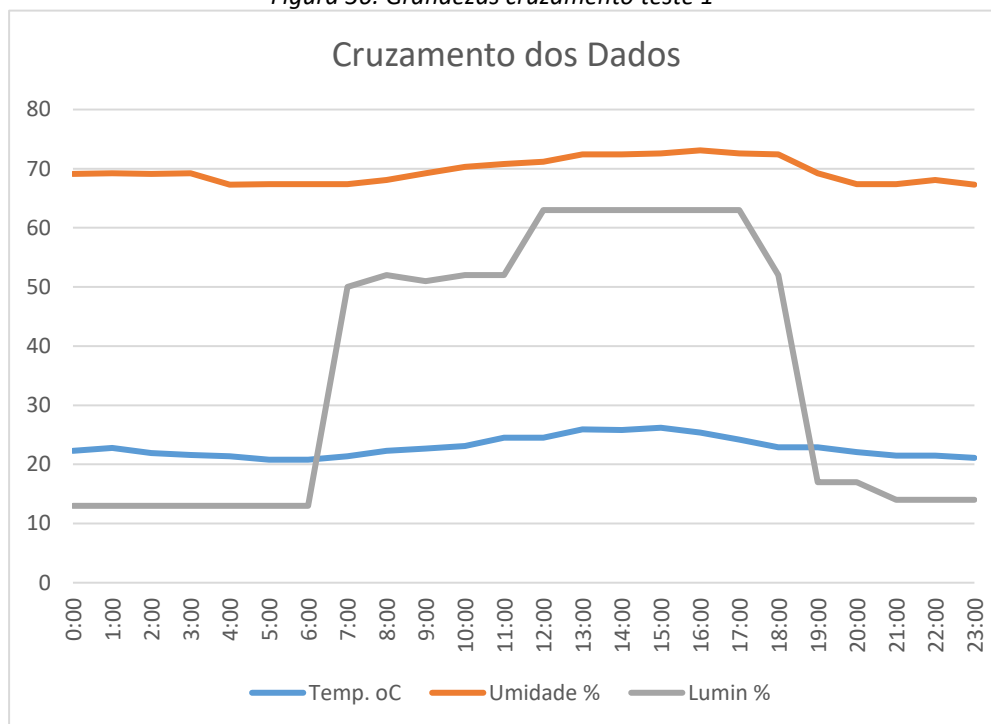
Data	Hora	Temp. oC	Umidade %	Lumin %
07/03/2021	00:00	22,3	69,1	13
07/03/2021	01:00	22,8	69,2	13
07/03/2021	02:00	21,9	69,1	13
07/03/2021	03:00	21,6	69,2	13
07/03/2021	04:00	21,4	67,3	13
07/03/2021	05:00	20,8	67,4	13
07/03/2021	06:00	20,8	67,4	13
07/03/2021	07:00	21,4	67,4	50
07/03/2021	08:00	22,3	68,1	52
07/03/2021	09:00	22,7	69,2	51
07/03/2021	10:00	23,1	70,3	52
07/03/2021	11:00	24,5	70,8	52
07/03/2021	12:00	24,5	71,2	63
07/03/2021	13:00	25,9	72,4	63
07/03/2021	14:00	25,8	72,4	63
07/03/2021	15:00	26,2	72,6	63
07/03/2021	16:00	25,4	73,1	63
07/03/2021	17:00	24,2	72,6	63
07/03/2021	18:00	22,9	72,4	52
07/03/2021	19:00	22,9	69,2	17
07/03/2021	20:00	22,1	67,4	17
07/03/2021	21:00	21,5	67,4	14
07/03/2021	22:00	21,5	68,1	14
07/03/2021	23:00	21,1	67,3	14

Fonte: Autores

Também foi extraído um gráfico para uma melhor visualização, podemos verificar sua alteração linear das curvas o que demonstra a estabilidade do ambiente conforme a Figura 36.

A partir destas variáveis, observa-se através de gráfico as variações das grandezas em diferentes períodos do dia, cruzando dados e verificando as alterações de temperatura, umidade e luminosidade em relação as fases do dia, os gráficos tem um comportamento linear que aponta o quanto o ambiente é controlado, por se tratar de uma área interna protegida das variações do clima.

Figura 36: Grandezas cruzamento teste 1



Fonte: Autores

No segundo teste efetuado, o dispositivo foi colocado em um ambiente externo, desprotegido, exposto as variações do clima no decorrer do dia.

Ficando ligado pelo mesmo período de 24 horas, coletando os dados para uma posterior análise.

Após o período de 24 horas, os dados coletados foram baixados do banco de dados *Thingspeak*, em um arquivo no formato. CSV, e tratados através do *software excel*.

Os dados coletados estão demonstrados na Tabela 7.

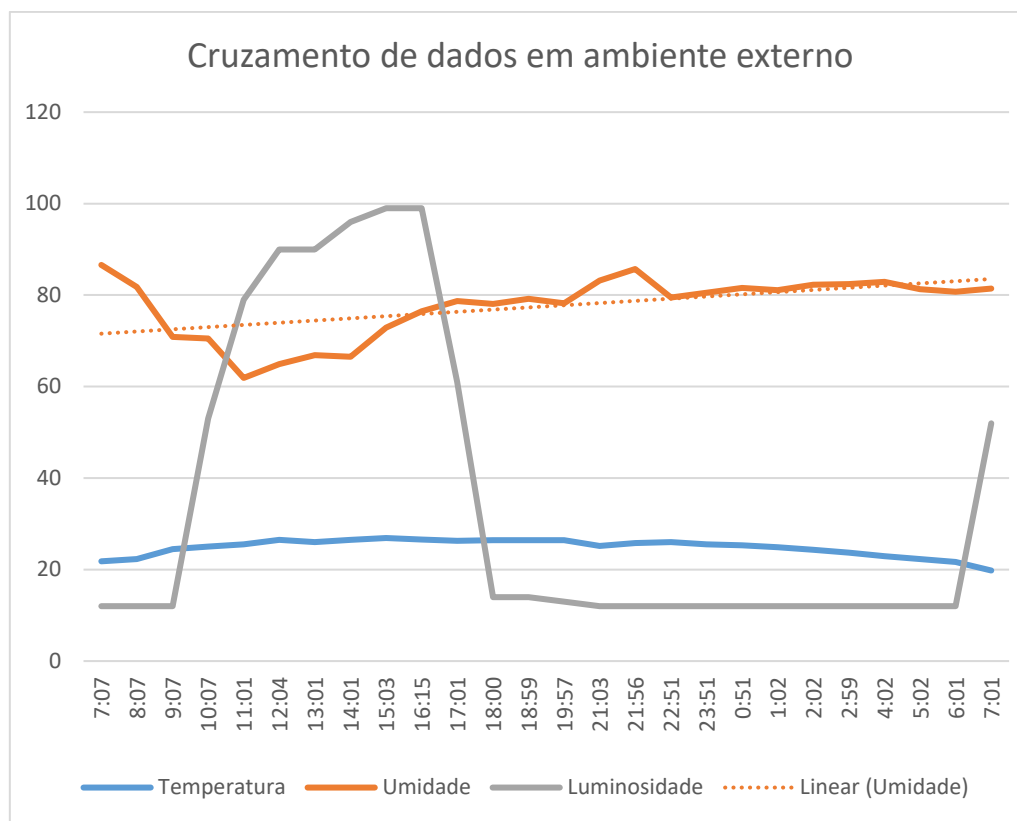
A partir destas variáveis, observa-se através do gráfico as variações das grandezas em diferentes períodos do dia, cruzando dados e verificando as alterações de temperatura, umidade e luminosidade em relação as fases do dia, os gráficos tem um comportamento não linear que aponta o quanto o ambiente é instável, por se tratar de uma área externa desprotegida das variações do clima, conforme a Figura 37.

Tabela 7: Grandezas coletadas teste 2

Data	Hora	Temperatura	Umidade	Luminosidade
12/04/2021	7:07	21,8	86,6	12
12/04/2021	8:07	22,3	81,8	12
12/04/2021	9:07	24,5	70,9	12
12/04/2021	10:07	25	70,5	53
12/04/2021	11:01	25,5	61,9	79
12/04/2021	12:04	26,5	64,9	90
12/04/2021	13:01	26	66,9	90
12/04/2021	14:01	26,5	66,5	96
12/04/2021	15:03	26,9	72,9	99
12/04/2021	16:15	26,6	76,5	99
12/04/2021	17:01	26,3	78,7	61
12/04/2021	18:00	26,4	78,1	14
12/04/2021	18:59	26,4	79,2	14
12/04/2021	19:57	26,4	78,2	13
12/04/2021	21:03	25,2	83,2	12
12/04/2021	21:56	25,8	85,7	12
12/04/2021	22:51	26	79,5	12
12/04/2021	23:51	25,5	80,5	12
13/04/2021	0:51	25,3	81,6	12
13/04/2021	1:02	24,9	81,1	12
13/04/2021	2:02	24,3	82,3	12
13/04/2021	2:59	23,7	82,4	12
13/04/2021	4:02	22,9	82,9	12
13/04/2021	5:02	22,3	81,3	12
13/04/2021	6:01	21,7	80,7	12
13/04/2021	7:01	19,8	81,4	52

Fonte: Autores

Figura 37: Grandezas cruzamento teste 2



Fonte: Autores

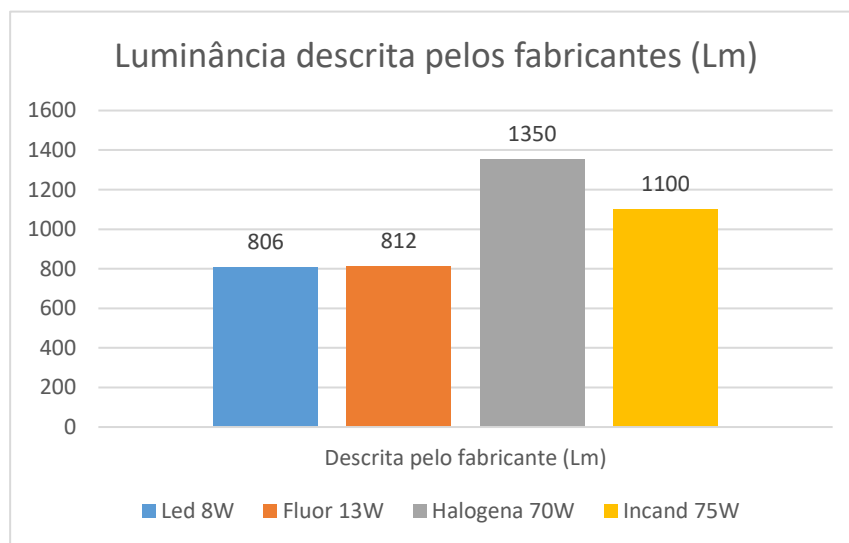
6.2 Testes Luminotécnicos.

O terceiro teste foi realizado em lâmpadas o qual se mostrou promissor para obter o levantamento de dados dos mesmos. Onde o equipamento foi colocado em um quarto com 7,37m², foram efetuados testes para verificar a equivalência, eficiência e nível de luminância entre os modelos de lâmpada mais comuns do mercado.

A luz que uma lâmpada irradia, relacionada à superfície à qual incide, define uma nova grandeza luminotécnica denominada de iluminamento, nível de iluminação ou iluminância.

Para determinar a eficiência energética de uma lâmpada devemos nos atentar a dois parâmetros, Lúmen (LM) e o Lux (LX). Lúmen é a unidade de medida do Fluxo luminoso, ou seja, a intensidade ou brilho da luz, e essa informação você encontra na embalagem. Já o Lux, é a quantidade de lúmens existentes em cada m², ou seja, $\text{Lux} = \text{Lúmen}/\text{m}^2$

Figura 38: Luminância dos fabricantes



Fonte: Autores

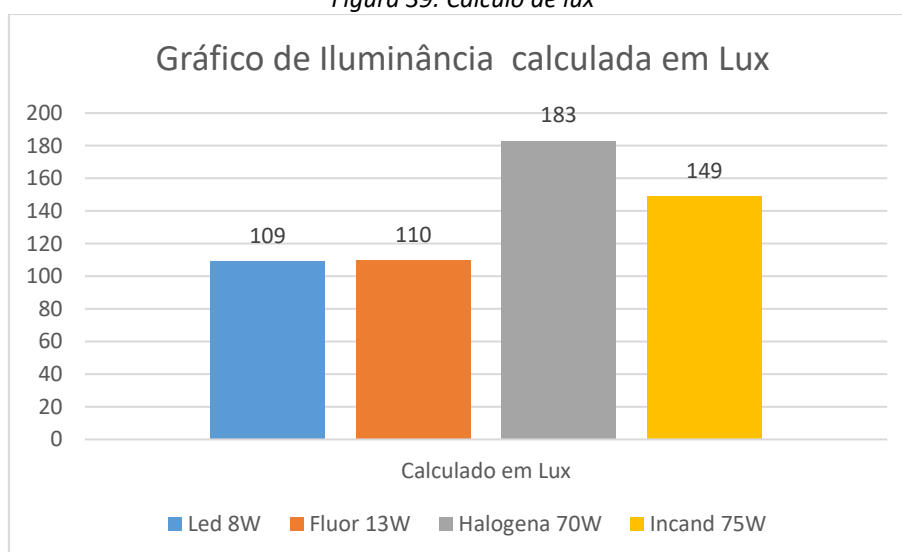
$$\text{Lâmpada de Led 8W: Lux} = \frac{Lm}{m^2} = \frac{806}{7,37} = 109 \text{ Lux.}$$

$$\text{Lâmpada Fluorescente 15W: Lux} = \frac{Lm}{m^2} = \frac{812}{7,37} = 110 \text{ Lux.}$$

$$\text{Lâmpada Halógena 70W: Lux} = \frac{Lm}{m^2} = \frac{1350}{7,37} = 183 \text{ Lux.}$$

$$\text{Lâmpada de Incandescente 75W: Lux} = \frac{Lm}{m^2} = \frac{1100}{7,37} = 149 \text{ Lux.}$$

Figura 39: Cálculo de lux



Fonte: Autores

Com base nestes números, calcula-se a eficiência energética de cada lâmpada assim para estes casos foram obtidos os seguintes valores:

$$\text{Lâmpada de Led 8W:} = \frac{Lm}{W} = \frac{806}{8} = 100,7 \text{ Lm/W}$$

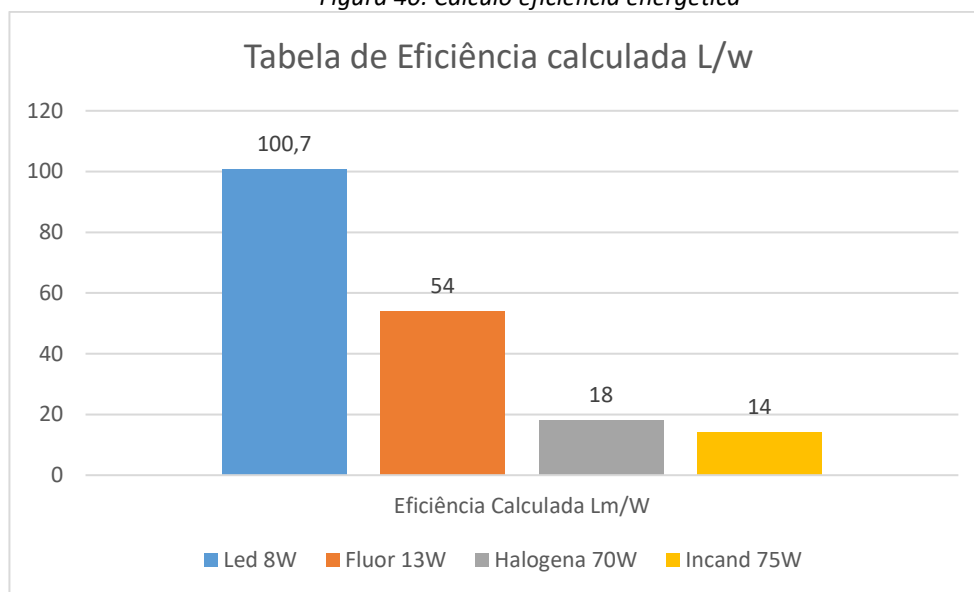
$$\text{Lâmpada Fluorescente 15W:} = \frac{Lm}{W} = \frac{812}{15} = 54 \text{ Lm/W}$$

$$\text{Lâmpada Halógena 70W:} = \frac{Lm}{W} = \frac{1350}{70} = 18 \text{ Lm/W}$$

$$\text{Lâmpada de Led 75W:} = \frac{Lm}{W} = \frac{1100}{75} = 14 \text{ Lm/W}$$

Estas grandezas foram representadas no gráfico conforme a Figura 40.

Figura 40: Cálculo eficiência energética

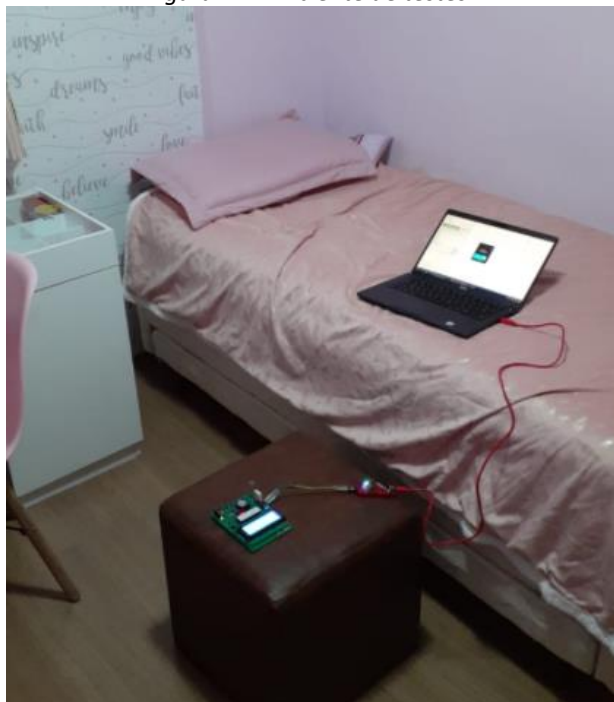


Fonte: Autores

Tomando como base de referência as grandezas calculadas o próximo passo foi iniciar os testes com o dispositivo para aferir se os valores teóricos se aproximavam dos práticos para certificar a eficácia do dispositivo para esta finalidade.

O primeiro valor coletado foi da lâmpada de Led 8W, em um ambiente interno um quarto de 7,37m², colocada a uma altura de 2,80m.

Figura 41: Ambiente de testes 2



Fonte: Autores

Os resultados dos testes das lâmpadas seguem nas Figuras 42, 43, 44 e 45 para cada tipo de lâmpada, LED, fluorescente, halógena e incandescente, respectivamente.

Figura 42: Teste LED



Fonte: Autores

Figura 43: Teste Fluorescente



Fonte: Autores

Figura 44: Teste Halógena



Fonte: Autores

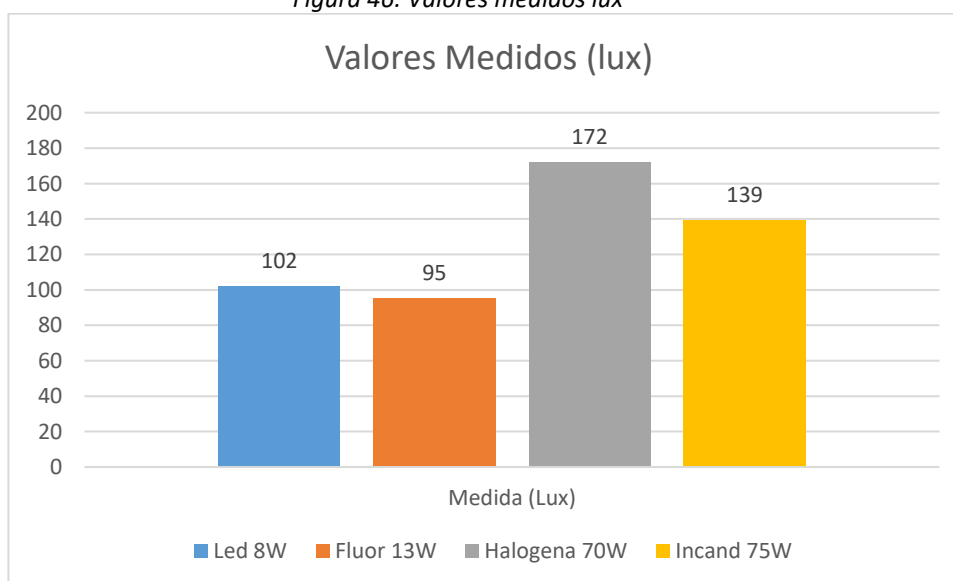
Figura 45: Teste Incandescente



Fonte: Autores

Após as medidas coletadas foram apresentados os valores do gráfico na Figura 46, para as quatro lâmpadas escolhidas para este experimento.

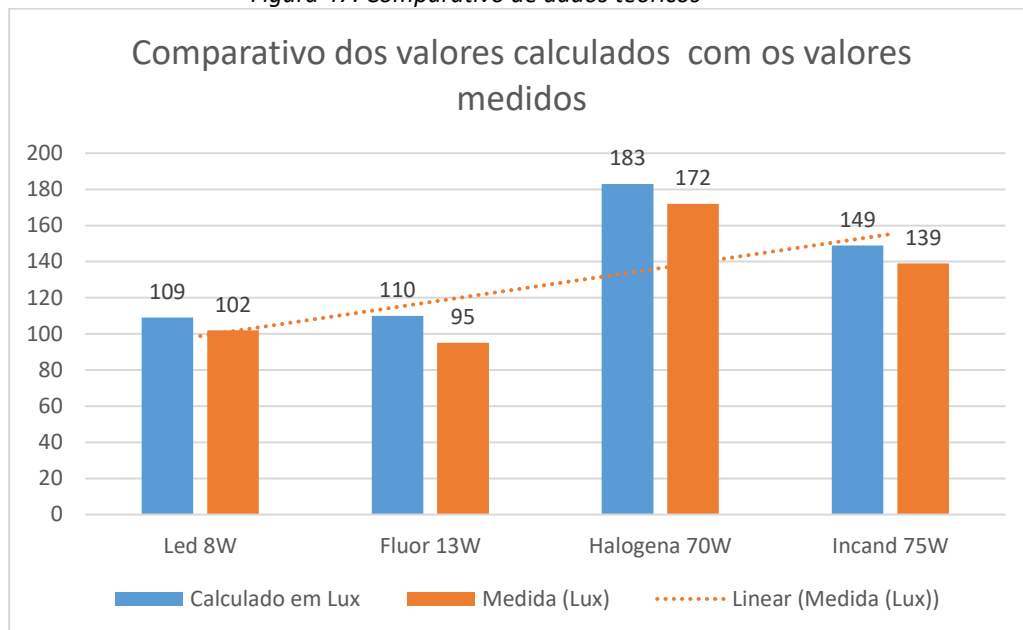
Figura 46: Valores medidos lux



Fonte: Autores

Cruzando os valores medidos e os valores calculados, agrupando-os em um gráfico, se obtêm-se o resultado do gráfico da Figura 47, que mostra em todas as amostras uma pequena queda nos valores medidos com os valores calculados.

Figura 47: Comparativo de dados teóricos



Fonte: Autores

Com base nestes valores foi possível calcular o valor real de luminância utilizando as fórmulas apresentadas.

Desenvolvendo os cálculos opostos para converter os valores de Lux para lúmen.

$$\text{Lâmpada de Led 8W: } Lm = lux * m^2 = 104 * 7,37 = 766,48 \text{ lm}$$

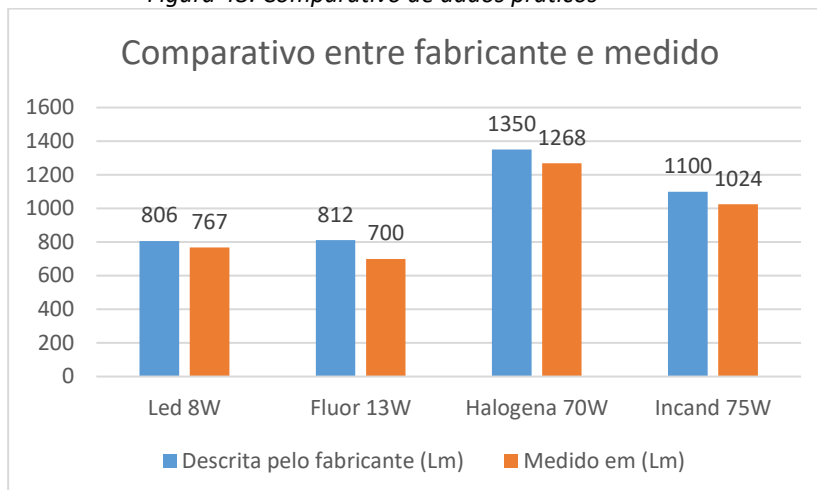
$$\text{Lâmpada Fluorescente 15W: } lux * m^2 = 95 * 7,37 = 700,15 \text{ lm}$$

$$\text{Lâmpada Halógena 70W: } lux * m^2 = 172 * 7,37 = 1267,64 \text{ lm}$$

$$\text{Lâmpada Incandescente 75W: } lux * m^2 = 139 * 7,37 = 1.024,43 \text{ lm}$$

Com estes dados é possível efetuar um comparativo entre os valores medidos e os valores calculados para obter uma comparação visual conforme Figura 48.

Figura 48: Comparativo de dados práticos



Fonte: Autores

Tendo os valores medidos em lúmen, é possível calcular a real eficiência luminosa de cada lâmpada, e aferir se seus valores estão dentro de uma faixa tolerável de diferença.

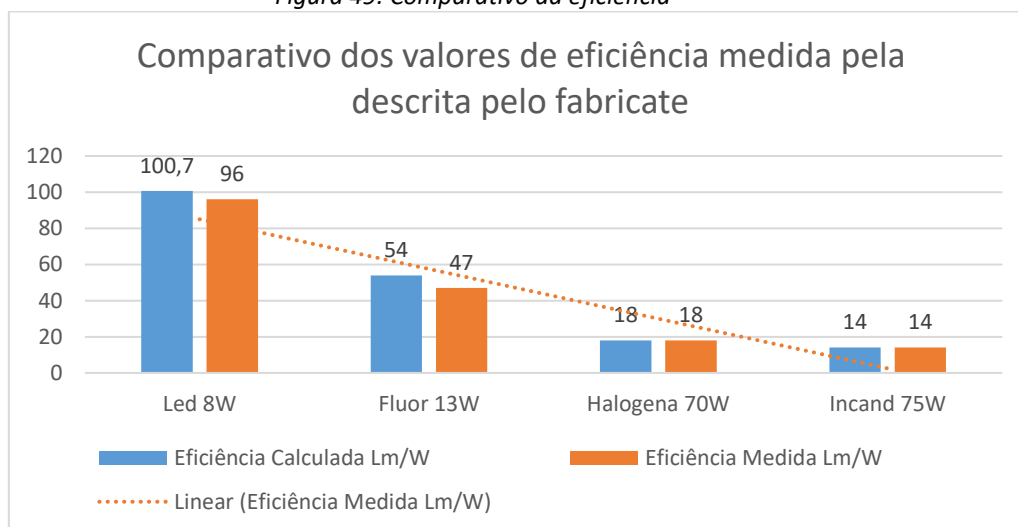
$$\text{Lâmpada de Led 8W: } = \frac{Lm}{W} = \frac{767}{8} = 96 \text{ Lm/W}$$

$$\text{Lâmpada Fluorescente 15W: } = \frac{Lm}{W} = \frac{700}{15} = 47 \text{ Lm/W}$$

$$\text{Lâmpada Halógena 70W: } = \frac{Lm}{W} = \frac{1268}{70} = 18 \text{ Lm/W}$$

$$\text{Lâmpada de Led 75W: } = \frac{Lm}{W} = \frac{1024}{75} = 14 \text{ Lm/W}$$

Figura 49: Comparativo da eficiência



Fonte: Autores

Com base nos dados coletados, os valores coletados e medidos ficaram bem próximos, sendo assim os testes se mostraram eficientes, para a finalidade de testes de eficácia das lâmpadas de mercado.

As imagens das Figuras 50 a 53 mostram as lâmpadas utilizadas para os testes, seguindo a referência de equivalência entre os modelos comparando luminosidade, potência e temperatura de funcionamento.

Na Figura 50 utilizou a lâmpada halógena de 70W - 127V – 1350 Lumens – 183 Lux.

Figura 50: Lâmpada halógena



Fonte: Autores

Na Figura 51 foi utilizada a lâmpada de Led de 8W – bivolt – 806 Lumens – 109 Lux.

Figura 51: Lâmpada LED



Fonte: Autores

Na Figura 52 foi utilizada a lâmpada fluorescente de 13W – 127V – 840 Lumens – 110 Lux.

Figura 52: Lâmpada fluorescente



Fonte: Autores

Na Figura 53 foi utilizada a lâmpada incandescente de 75W – 127V – 1100 Lumens – 149 Lux.





Figura 53: Lâmpada incandescente



Fonte: Autores

Para as escolhas das lâmpadas foi utilizada a Figura 54 de equivalência de potências das lâmpadas de mercado para selecionar as lâmpadas mencionadas para ter uma equivalência na luminosidade e na potência.

Figura 54: Equivalência de potências

EFICIÊNCIA		Menos				Mais			
TIPO									
		COMUM	HALÓGENA	CFL	LED				
CONSUMO		40 W	28 W	8 W	4 W				
		60 W	42 W	12 W	6 W				
		75 W	53 W	15 W	8 W				
		100 W	70 W	20 W	10 W				
DURABILIDADE		1 ano	1-3 anos	6-10 anos	15-25 anos				
ECONOMIA		×	até 30%	até 80%	até 95%				

Fonte: <https://blogligacaohomecenter.wordpress.com/2015/07/08/qual-a-diferenca-entre-lampada-incandescente-fluorescente-e-led/> acessado 10/04/2021

Os resultados foram satisfatórios, pois todas as lâmpadas apresentaram intensidade luminosa bem próximas, no critério eficiência energética, a lâmpada de led de 8W calculando sua eficiência Lm/W lumens por watts apresentou uma eficiência de 96 Lm/W, seguida da lâmpada fluorescente de 15W que apresentou uma eficiência de 47 Lm/W, apresentaram melhores resultados em luminosidade por potencia consumida.

Já a lâmpada incandescente de 75W, apresentou uma eficiência de 14 Lm/W e a lâmpada halógena de 70W apresentou uma eficiência de 14 Lm/W, apresentaram uma baixa eficiência energética, iluminando menos pela potência consumida.

Com base nos dados coletados por este dispositivo, desenvolvido neste trabalho de conclusão de curso podemos atestar a eficiência e comparar com o descrito pelo *datasheet* das lâmpadas.

6.3 Limitações encontradas e conclusão parcial

As principais limitações encontradas durante o desenvolvimento do trabalho estão relacionadas ao *software* de desenvolvimento que apresentam limitações nas versões *Demo* utilizadas para o desenvolvimento do projeto.

Esta limitação em vários momentos nos impediu de explorar todo potencial do *hardware*, limitando assim a funcionalidade do dispositivo que poderia ir além dos propósitos mostrados neste projeto.

Outra dificuldade encontrada foi o desenvolvimento do *hardware*, devido a complexidade para atender nossas expectativas de tamanho e aplicabilidade necessitamos de uma placa dupla face, com trilhas ilhas e espaçamentos mínimos possíveis, exigindo na configuração da placa, uma tecnologia que dificultava a obtenção do protótipo de forma artesanal. Na indústria nacional não encontramos nenhuma empresa que desenvolvesse o protótipo, somente aceitando pedidos mínimos se 100 peças, o que nos obrigou a produzir nosso protótipo na china, o que elevou os custos e o prazo de entrega.

7 CONCLUSÃO

O protótipo conseguiu atender com algumas limitações aquilo a qual foi programado. Conseguimos colocar os sensores de luminosidade, temperatura e umidade em perfeito funcionamento, coletando os dados e apresentando eles em *display* para um acompanhamento local.

Outro ponto satisfatório é a comunicação do microcontrolador com o módulo Wi-Fi, o que possibilita um acesso remoto de todos os dados coletados, enviados através de tecnologia *IoT* para a nuvem, sendo armazenados e apresentados em plataforma através do site www.thingspeak.com.

Em conformidade com nossos objetivos gerais, conseguimos obter um dispositivo funcional e que atende ao que foi proposto, definindo assim um resultado satisfatório.

Em conformidade aos objetivos específicos conseguimos obter dados coerentes e relevantes com os dados comparados, sendo assim obtivemos resultados satisfatórios.

8 PROJETOS FUTUROS

No decorrer do desenvolvimento e durante os testes, foi observado que alguns recursos podem ser implementados para tornar o sistema mais autônomo, apresentando-os:

- Acionamento de sistema de ventilação, em caso de aumento da temperatura.

- Acionamento de iluminação artificial e aumento de eficiência luminosa e energética.

- Desenvolvimento de um sistema para supervisionar de maneira autônoma a queda da qualidade luminosa do ambiente.

- O desenvolvimento de uma interface móvel para monitoramento e acionamento remoto dos reles.

- Também será importante agregar inteligência ao sistema, permitindo-lhe tomar decisões em atividades críticas.

- Possibilitar ao sistema o controle de entidades mais complexas como um sistema de detecção e combate a incêndio, com o acionamento de alarmes.

9 REFERÊNCIAS BIBLIOGRÁFICAS

[1] Artigo: Resultados Procel 2015 ano base 2014

http://www.procelinfo.com.br/resultadosprocel2015/docs/rel_procel2015_web.pdf?1

<Acessado em 26/05/2019>

[2] Artigo: Proteste

<https://www.proteste.org.br/institucional/imprensa/press-release/2016/teste-com-lampadasled-aponta-qualidade-duvidosa-em-sete-marcas>

< Acessado em 26/05/2019 >

[3] Apostila: Eletrônica – Automação Industrial

PAREDE, I. M.; GOMES, L. E. L.; HORTA, E.; SILVA, L. C. C.; SUZUKI, J.
Manual Técnico Centro Paula Souza. São Paulo: Fundação Padre Anchieta, 2011. 227p

[4] Artigo: Embarcados – Microchip Microcontroladores PIC de 8 bits

<https://www.embarcados.com.br/pic/>

<Acessado em 23/11/2020>

[5] Artigo: Embarcados – Apresentando o módulo ESP8266

<https://www.embarcados.com.br/modulo-esp8266/>

<Acessado em 21/11/2020>

[6] Artigo: FILIPEFLOP – Sensor de umidade e temperatura AM2302 DHT22

<https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/>

<Acessado em 26/11/2020>

.

[7] Artigo: LDR – O que é e como funciona

<http://mundoprojetado.com.br/ldr-o-que-e-e-como-funciona/>

<Acessado em 14/10/2020>

[8] Artigo: Curso de Eletrônica – O que é um LDR (Light Dependent Resistor)

<http://www.bosontreinamentos.com.br/eletronica/curso-de-eletronica/curso-de-eletronica-o-que-e-um-ldr-light-dependent-resistor/>

<Acessado em 14/10/2020>

[9] Artigo: FILIPEFLOP – Real Time Clock RTC DS1307

<https://www.filipeflop.com/produto/real-time-clock-rtc-ds1307/>

<Acessado em 18/11/2020>

[10] Artigo: Curto Circuito – Display LCD 20X4 Backlight Azul

<https://www.curtocircuito.com.br/display-lcd-20x4-backlight-azul.html>

<Acessado em 16/11/2020>

[11]Artigo: MicroElectronics

<https://www.mikroe.com/mikroc>

< Acessado em 11/05/2019 >

[12]Artigo: Labcenter

<https://www.labcenter.com/>

< Acessado em 11/05/2019 >

ANEXO A: Código Fonte

```

/*****
/***** Projeto Final *****/
/*****
//          Grupo: Milton Luiz Guetti Junior 16202671
//          Thiago Rezende da Silva 16202633
//          Thiago Nogueira de Chaves 16202526
/*****
/*DESENVOLVER UM SISTEMA UTILIZANDO MIKROC PARA IMPLEMENTAÇÃO DE UM
PROJETO UTILIZANDO UM LCD (20X4) COM MICROCONTROLADOR PIC 18F4520 COM CRISTAL
DE 8MHZ*/
/*****
/*****
/***** Datada da Última atualização *****/
/***** 23/11/2020 18:50 *****/
/*****
/***** Timer 0 *****/
/*prescaler configurado para 1:16
ciclo de máquina = 1us
estouro do TMR0 1us x 16 x 256 = 4,96ms*/
/***** Equação do Timer1 *****/
// Os segundos são incrementados a cada estouro do TMR1
// prescaler configurado para 1:8
// TMR1 conta até 62500
// ciclo de máquina = 1us
// Estouro = TMR1 x prescaler x ciclo de máquina
// = 62500 x 8 x 1E-6
// =500ms
/*****
/***** PINAGEM DO DISPLAY *****/
/*****
sbit LCD_RS at RD2_bit;
sbit LCD_EN at RD3_bit;
sbit LCD_D4 at RD4_bit;
sbit LCD_D5 at RD5_bit;
sbit LCD_D6 at RD6_bit;
sbit LCD_D7 at RD7_bit;
sbit LCD_RS_Direction at TRISD2_bit;
sbit LCD_EN_Direction at TRISD3_bit;
sbit LCD_D4_Direction at TRISD4_bit;
sbit LCD_D5_Direction at TRISD5_bit;
sbit LCD_D6_Direction at TRISD6_bit;
sbit LCD_D7_Direction at TRISD7_bit;
// pinagem I2C
sbit Soft_I2C_Scl at RC3_bit;
sbit Soft_I2C_Sda at RC4_bit;
sbit Soft_I2C_Scl_Direction at TRISC3_bit;
sbit Soft_I2C_Sda_Direction at TRISC4_bit;

```

```

// pinagem DHT
sbit DataDHT at PORTC.B5; //Sinal de dados provenientes do sensor
sbit InDataDHT at TRISC.B5;
/*****
/***** Variaveis Globais *****/
/*****
//variáveis leitura Temperatura e umidade
unsigned char Check, T_byte1, T_byte2, RH_byte1, RH_byte2, Ch;
unsigned Temp, RH, Sum;
unsigned be, me, ne, Sum2, h1, h2, h3;
//variáveis LDR

char LDR[6];
int botao = 0;
char valorAD0;

//variáveis do Relogio
char *text = "00:00:00"; //Ponteiro para texto do relógio no LCD
char *text2 = "00/00/00"; //Ponteiro para texto da data
char dias = 0x00, mes = 0x00, ano = 0x00; //Variáveis data
char horas = 0x00, minutos = 0x00, segundos = 0x00, date = 0x00; //variáveis Hora
//variáveis de ajuste do relógio
char flags = 0x00;
char control_adj_watch = 0x05; //Controle do ajuste do relógio
/*****
/***** Mapeamento de Hardware *****/
/*****
#define B0 portb.rb0 //Botão para incrementar as unidades
#define B1 portb.rb1 //Botão para incrementar as dezenas
#define B2 portb.rb2 //Botão para ajuste das horas
#define B3 Portb.rb3 //Botão B3
#define LED1 portb.rb4 //Define Led1
#define LED2 portb.rb5 //Define Led2
#define LED3 portb.rb6 //Define Led3
#define TESTE portb.rb7 //Define Porta para teste de pulso
/*****
/***** Declaração das funções *****/
/*****
void inicia_MCU(); //função inicia setup do microcontrolador
void inicia_display(); //função para inicialização do display
void Leitura_DHT(); //função que efetua a leitura do sensor DHT22
void Leitura_lDr(); //função que faz a leitura do sensor de umidade
void disp_horario(); //Função para mostrar as horas
void relógio(); //Função de controle das variáveis do relógio
void ajuste_do_relogio(); //Função para ajusta do relógio
void principal(); //função principal
void temperatura(); //função que mede a temperatura
void luminosidade(); //função que mede a luminosidade
void umidade(); //função que mede a umidade
void inicia_dipositivo(); //função que verifica o funcion do display e dos leds
void ajusta_relogio(); //Função para ajuste de tempo

```

```

void uart(); //Função comunicação
/*****
/***** Inicia Display *****/
/*****
void inicia_display(){
    Lcd_Init(); //inicia display
    Lcd_Cmd(_LCD_CLEAR); //Limpa display
    Lcd_Cmd(_LCD_CURSOR_OFF); //Apaga Display
}
/*****
/***** Inicia MCU *****/
/*****
void inicia_MCU(){
    ADCON0 = 15;
    ADCON1 = 15;
    TRISA0_bit = 1;
    TRISB = 0b00001111; //Define portb como entrada
    PORTB = 0b00001111;
    TRISD = 0; //Configura portd como saída
    PORTD = 0; //Define nível lógico 0 para portd
    TRISC = 1;
    PORTC = 0;
// configuração do timer 1
    CMCON = 0x07; //Desabilitar os comparadores
    T1CON = 0b00001001; //0011 1001 Prescaler 1:1 TMR1, TMR1 on
        // 0 bit7: RD16 -> Habilita modo de 8 bits
        // 0 bit6: T1RUN -> Clock a partir de outra fonte
        // 0-0 bit5-4: T1CKPS<1:0> -> Prescaler 1:1
        // 1 bit 3: T1OSCEN -> Oscilador do Timer1 habilitado
        // 0 bit 2: T1SYNC -> Clock síncrono
        // 0 bit 1: TMR1CS -> Timer1 incrementa com clock externo
        // 1 bit 0: TMR1ON -> Liga contagem do Timer1
    TMR1L = 0x271; //Inicializamos o TMR1 em 64536 contagem de 1000
    TMR1H = 0x7D1;
    GIE_bit = 0x01; //Habilita a interrupção global
    PEIE_bit = 0x01; //Habilita a interrupção por periféricos
    TMR1IE_bit = 0x01; //Habilita a interrupção do TMR1
    UART1_Init(9600); // Inicializa a comunicação UART a 9600bps
    Soft_I2C_Init(); // Inicializa a comunicação I2C
    delay_ms(100); //Aguarda 100 milisegundos para estabilizar
}
/*****
/***** Interrupção TMR1 *****/
/*****
void interrupt(){
    if(TMR1IF_bit){ //Houve o estouro do TMR1?
        TMR1IF_bit = 0x00; //Limpa a flag
    }
} //fecha void interrupt
/*****
/***** função Uart *****/

```

```

/*****/
void uart(){
    UART1_Write_Text(text2); //envia Data via UART
    UART1_Write(32); // envia o comando de espaço
    UART1_Write_Text(text); //envia a hora via UART
    UART1_Write(32); // envia o comando espaço
    UART1_Write(be); // envia decimal da leitura DHT temperatura via UART
    UART1_Write(me); // envia unidade da leitura DHT temperatura via UART
    UART1_Write_Text("."); // envia o texto ponto
    UART1_Write(ne); // envia unidade da leitura temperatura via UART
    UART1_Write_Text("oC"); // envia o texto graus
    UART1_Write(32); // envia o comando de espaço
    UART1_Write(h1); // envia decimal da humidade
    UART1_Write(h2); // envia unidade da umidade
    UART1_Write_Text("."); // envia texto ponto
    UART1_Write(h3); //envia unidade humidade
    UART1_Write(37); //envia texto porcentagem
    UART1_Write_Text(LDR); // envia dados luminosidade
    UART1_Write(37); //envia texto porcentagem
    UART1_Write(13); //Quebra de linha
}
/*****/
/***** FUNÇÃO RELOGIO *****/
/*****/
void relógio(){
    Leitura_DHT(); //Executa a leitura do sensor DHT para Umidade e temperatura
    Leitura_Ldr(); //Executa a leitura do sensor LDR Para luminosidade.
    Uart(); //Executa o envio de dados via UART

    Soft_i2c_start();
    Soft_i2c_write(0xD0);
    Soft_i2c_write(0x00);
    Soft_i2c_Start();
    Soft_i2c_write(0xD1);
    segundos = Bcd2Dec(Soft_i2c_read(1));
    minutos = Bcd2Dec(Soft_i2c_read(1));
    horas = Bcd2Dec(Soft_i2c_read(1));
    date = Bcd2Dec(Soft_i2c_read(1));
    dias = Bcd2Dec(Soft_i2c_read(1));
    mes = Bcd2Dec(Soft_i2c_read(1));
    ano = Bcd2Dec(Soft_i2c_read(0));
    Soft_i2c_stop();
}
void write_time(){
    Soft_i2c_stop();
    Soft_i2c_start();
    Soft_i2c_write(0xD0);
    Soft_i2c_write(0x00);
    Soft_i2c_write(Dec2Bcd(00)); //segundos
    Soft_i2c_write(Dec2Bcd(44)); //minutos
}

```

```

        Soft_i2c_write(Dec2Bcd(19)); //horas
        Soft_i2c_write(Dec2Bcd(27)); //dia da semana
        Soft_i2c_write(Dec2Bcd(23)); //data
        Soft_i2c_write(Dec2Bcd(11)); //mes
        Soft_i2c_write(Dec2Bcd(20)); //ano
        Soft_i2c_stop();
    }
void disp_horario(){

    text[7] = segundos%10 + '0';
    text[6] = segundos/10 + '0';
    text[4] = minutos%10 + '0';
    text[3] = minutos/10 + '0';
    text[1] = horas%10 + '0';
    text[0] = horas/10 + '0';
    text2[1] = dias%10 + '0';
    text2[0] = dias/10 + '0';
    text2[4] = mes%10 + '0';
    text2[3] = mes/10 + '0';
    text2[7] = ano%10 + '0';
    text2[6] = ano/10 + '0';

    Lcd_Out(1,13,text);          //Imprimi o horário no display
    Lcd_Out(1,1,text2);         //Imprimi a data no display
}
/*****
/***** FUNÇÃO LEITURA DA LUMINOSIDADE *****/
/*****/

void Leitura_Ldr(){
    while(1){
        valorAD0 = ADC_Get_Sample(0)/623;
        WordToStr(valorAD0,LDR);
        break;
    }
}

/*****
/***** FUNÇÃO DHT22 *****/
/***** LEITURA DE UMIDADE E LUMINOSIDADE *****/
/*****/

char ReadData(){
    char i, j;
    for(j = 0; j < 8; j++){
        while(!PORTC.F5); //Enquanto PORTC.F5 estiver em nivel alto
            delay_us(30); //Aguarde 30 microssegundos
        if(PORTC.F5 == 0) //Se portc.rc5 for igual a zero
            i&= ~(1<<(7 - j)); //Limpe bit (7-b)
        else {i|= (1 << (7 - j)); //Seta bit (7-b)
            while(PORTC.F5);} //Enquanto portc.F5 estiver em nivel baixo
        }
    return i;
}

```

```

void Leitura_DHT() {
    ADCON1=0x06;
    while(1){
        TRISC.F5 = 0; //Configura RC5 como saida
        DataDHT = 0; //define nivel logico baixo para a entrada
        delay_ms(18); //aguarda 18 milissegundos
        DataDHT = 1; //atribui nivel logico alto para a entrada
        delay_us(30); //aguarda 30 microsegundos
        TRISC.F5 = 1; //configura RC5 como entrada
        Check = 0; //atribui 0 para a variável Check
        delay_us(40); //aguarda 40 microssegundos
        if (PORTC.F5 == 0){ //verifica se portc.rc5 = 0
            delay_us(80); //aguarda 80 microssegundos
            if (PORTC.F5 == 1) Check = 1; delay_us(40);} //Se portc.rc5 = 1
        atribua 1 a variavel check
        if(Check == 1){
            RH_byte1 = ReadData();
            RH_byte2 = ReadData();
            T_byte1 = ReadData();
            T_byte2 = ReadData();
            Sum = ReadData();
            if(Sum == ((RH_byte1+RH_byte2+T_byte1+T_byte2) & 0XFF)){
                Temp = T_byte1;
                Temp = (Temp << 8) | T_byte2;
                RH = RH_byte1;
                RH = (RH << 8) | RH_byte2;
                if (Temp > 0X8000){
                    Temp = Temp & 0X7FFF;
                }
                be =(48 + ((Temp / 100) % 10));
                me = (48 + ((Temp / 10) % 10));
                ne = 48 + (Temp % 10);
                h1 = 48 + ((RH / 100) % 10);
                h2 = 48 + ((RH / 10) % 10);
                h3 = 48 + (RH % 10);
            } //fecha if sun
            else{
                Lcd_Cmd(_LCD_CLEAR);
                Lcd_Out(1, 1, "error");
            } //fecha else
            delay_ms(1000);
        }
        break;
    } //fecha while
} //fecha void leitura dht
}

/*****
/***** Chama a função principal *****/
/*****/

void main(){ //função principal
    inicia_MCU();
    inicia_display();
}

```

```

        inicia_dipositivo();
while(1){
    principal();
    ajusta_relogio();
    temperatura();
    luminosidade();
    umidade();
}
} //fecha void main

/*****
/*****                               Menu Principal                               *****/
/*****

void principal(){
    delay_ms(500);
    while(B3 == 1){
        relógio();
        disp_horario();           //Função para exibir a hora atual
        Lcd_Out(2,1,"Temp Umid Lumi");
        Lcd_Chr(3,1,be);
        Lcd_Chr(3,2,me);
        Lcd_Out_Cp(".");
        Lcd_Chr(3,4,ne);
        Lcd_Out_Cp("oC");
        Lcd_Chr(3,9,h1);
        Lcd_Chr(3,10,h2);
        Lcd_Out_Cp(".");
        Lcd_Chr(3,12,h3);
        Lcd_Out_Cp("%");
        Lcd_Out(3,14,LDR);
        Lcd_Out_Cp("%");
        Lcd_Out(4,7,"B3:Menu");
    }
    Lcd_Cmd(_LCD_CLEAR);
}

/*****
/*****                               inicializa dispositivo                               *****/
/*****

void inicia_dipositivo(){
    Lcd_Out(1,1,"Hello World");
    LED1 = 1;
    LED2 = 1;
    LED3 = 1;
    delay_ms(1000);
    LED1 = 0;
    LED2 = 0;
    LED3 = 0;
    Lcd_Cmd(_LCD_CLEAR);
}

/*****
/*****                               FUNÇÃO AJUSTA RELOGIO                               *****/

```

```

/*****
void ajusta_relogio(){          //Função para ajuste do relógio
    delay_ms(500);
    while(B3 == 1){
        disp_horario();          //Função para mostrar as horas
        Lcd_Out(3,3,"Ajusta Relogio");
    }
    Lcd_Cmd(_LCD_CLEAR);
} //end ajusta_relogio
/*****
/*****      menu_Temperatura      *****/
/*****
void temperatura(){
    delay_ms(500);
    while(B3 == 1){
        Lcd_Out(1,1,"Temperatura");
        Lcd_Out(2,1,"Temp  Min  Max");
        Lcd_Chr(3,1,be);
        Lcd_Chr(3,2,me);
        Lcd_Out_Cp(".");
        Lcd_Chr(3,4,ne);
        Lcd_Out_Cp("oC");
        Lcd_Out(3,8,"20.0oC");
        Lcd_Out(3,16,"40.0oC");
        Lcd_Out(4,1,"B1:min+ B2:max+ B3:^");
    } //fecha while
    Lcd_Cmd(_LCD_CLEAR);
} //fecha o void menu_Temperatura
/*****
/*****      menu_luminosidade      *****/
/*****
void luminosidade(){
    delay_ms(500);
    while(B3 == 1){
        Lcd_Out(1,1,"Luminosidade");
        Lcd_Out(2,1,"Lum  Min  Max");
        Lcd_Out(3,1,LDR);
        Lcd_Out_Cp("%");
        Lcd_Out(4,1,"B1:min+ B2:max+ B3:^");
    } //fecha while(1)
    Lcd_Cmd(_LCD_CLEAR);
} //fecha o void Lumin_menu
/*****
/*****      menu_Umidade      *****/
/*****
void umidade(){
    delay_ms(500);
    while(B3 == 1){
        Lcd_Out(1,1,"Umidade");
        Lcd_Out(2,1,"Umi.  Min.  Max.");
        Lcd_Chr(3,1,h1);

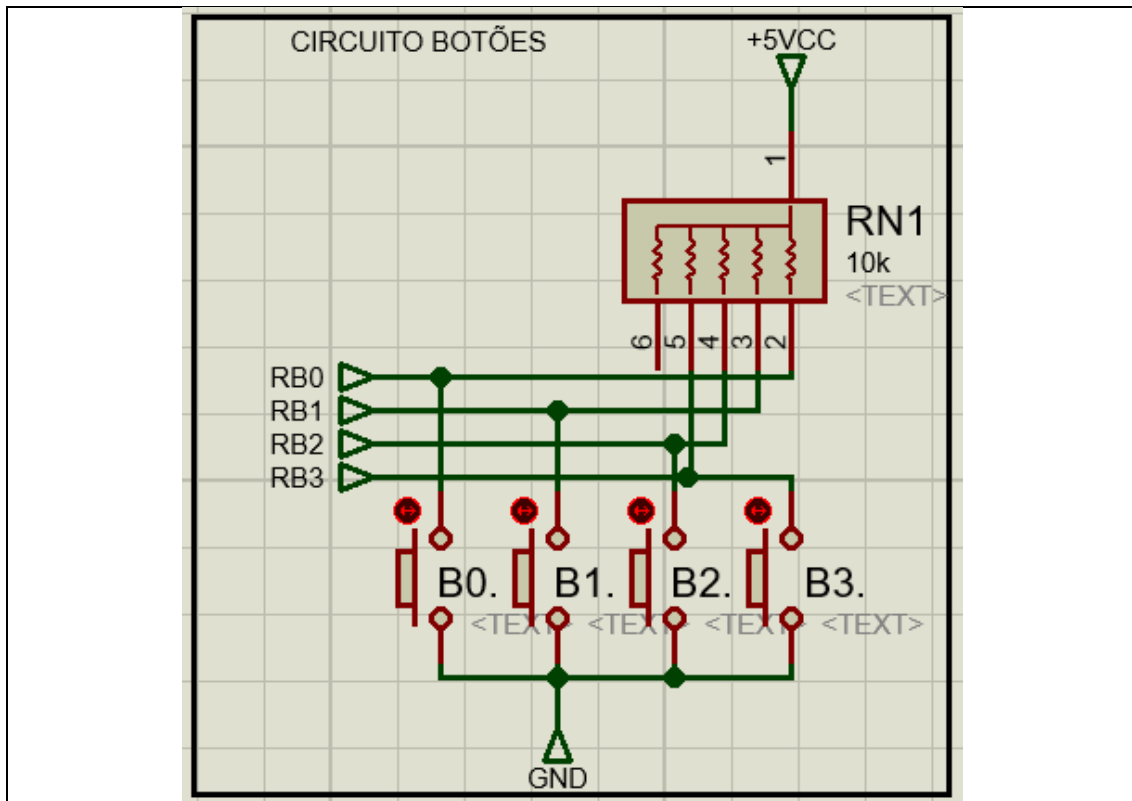
```



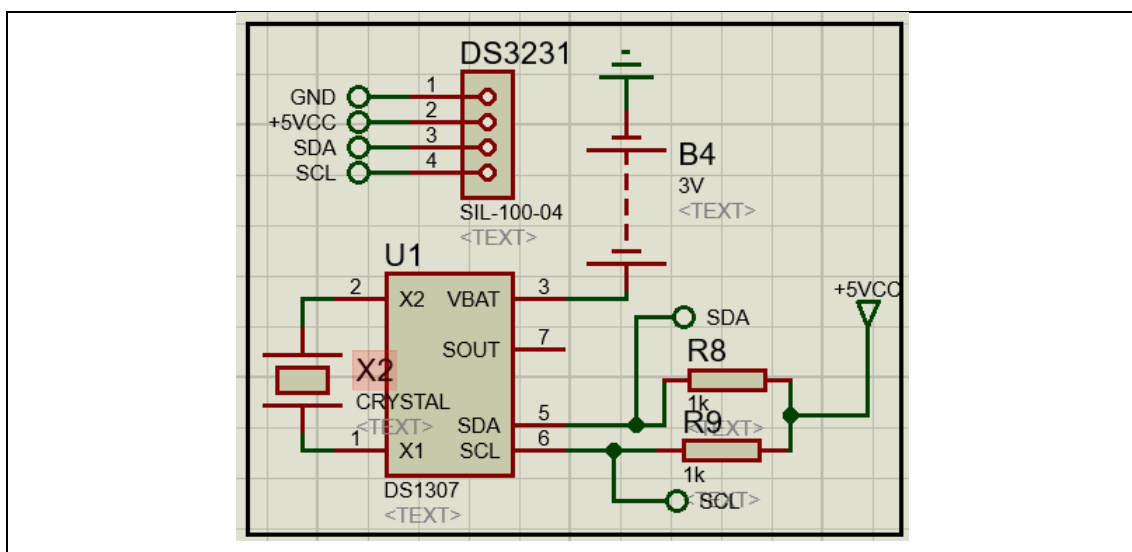
```
Lcd_Chr(3,2,h2);
Lcd_Out_Cp(".");
Lcd_Chr(3,4,h3);
Lcd_Out_Cp("%");
Lcd_Out(3,8,"20.0");
Lcd_Out_Cp("%");
Lcd_Out(3,16,"80.0");
Lcd_Out_Cp("%");
Lcd_Out(4,1,"B1:min+ B2:max+ B3:^");
}
Lcd_Cmd(_LCD_CLEAR);
} // fecha o void Lumin_menu
```

ANEXO B: Circuitos Elétricos

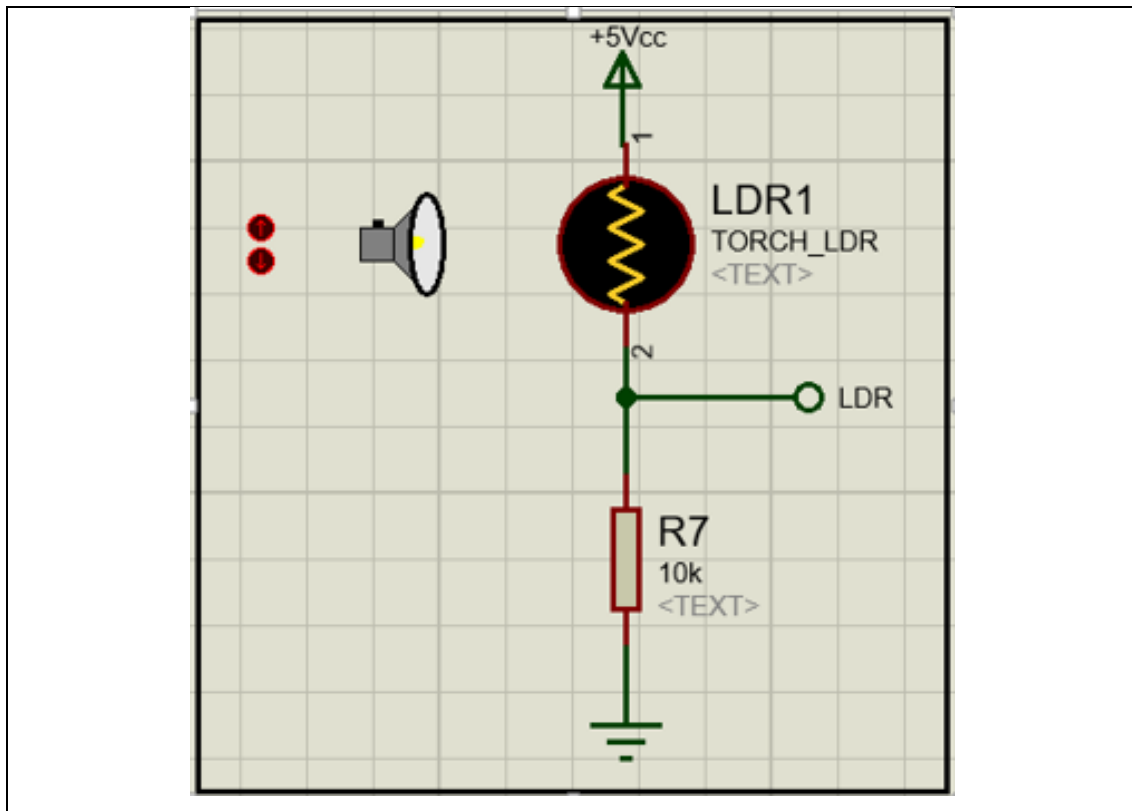
- Circuito de botões para ajustes no dispositivo.



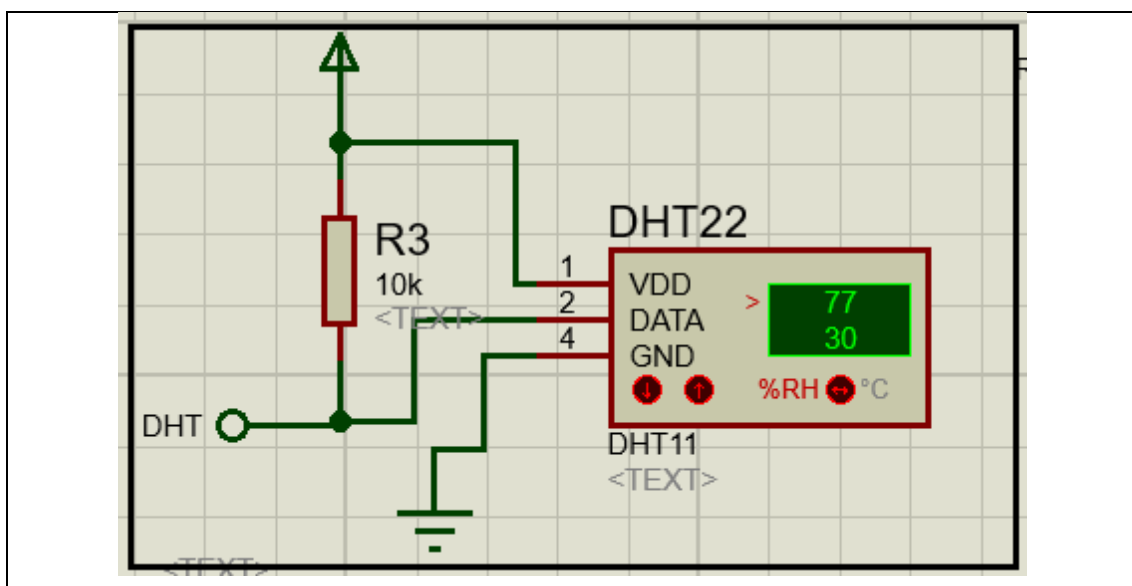
- Circuito de controle do relógio;



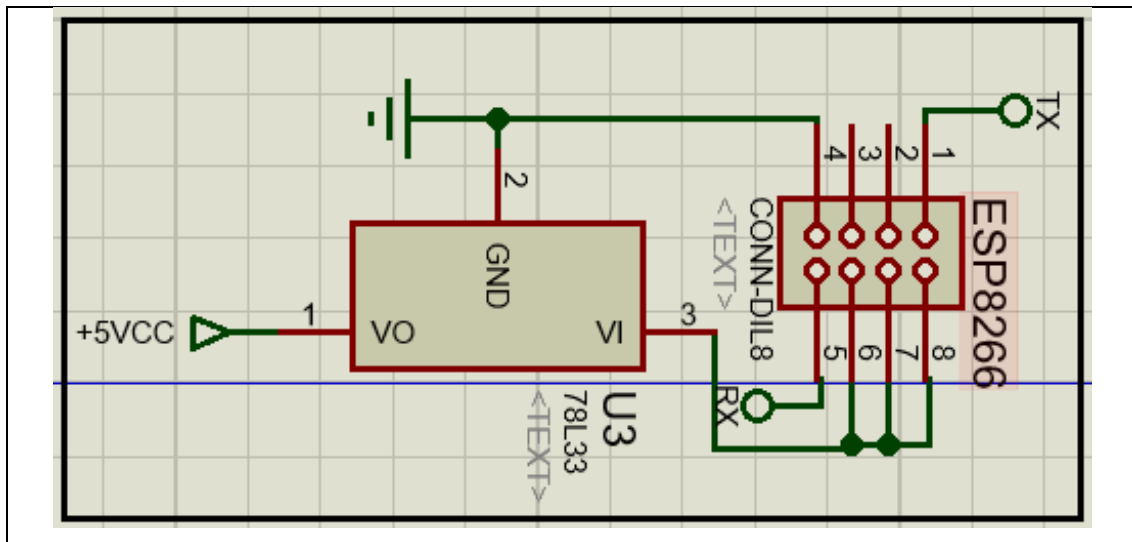
- Circuito acionamento luminosidade;



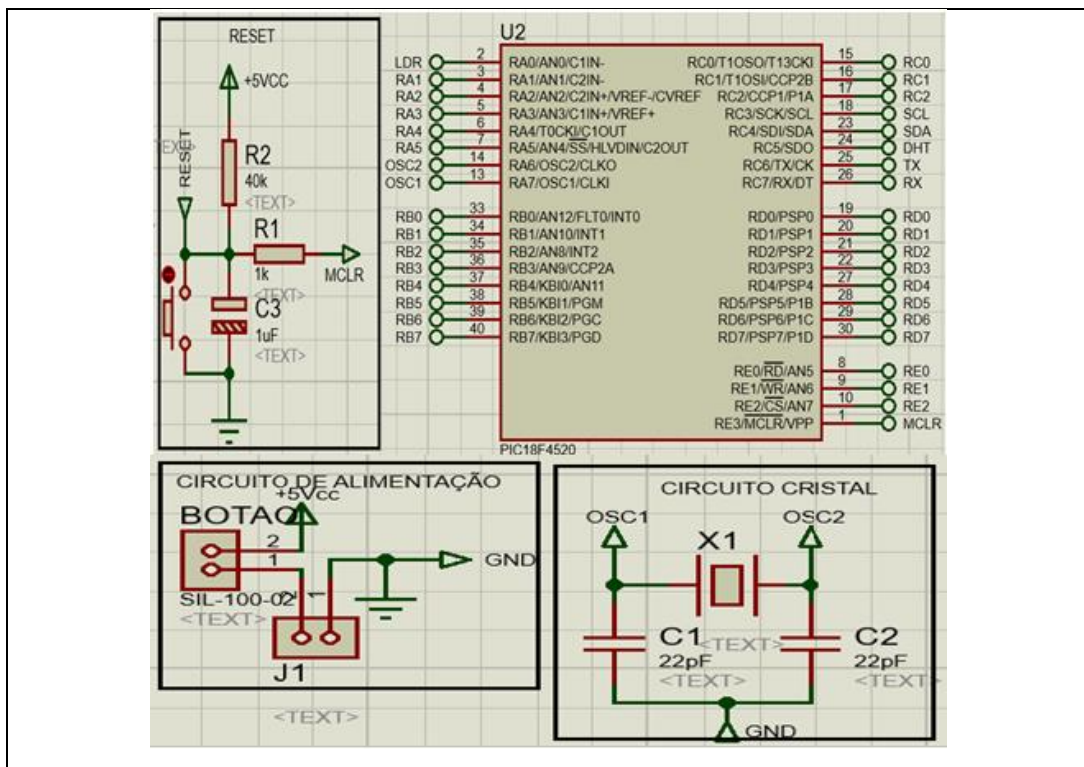
- Circuito de controle do sensor de temperatura e umidade;



- Circuito de controle de comunicação;



- Circuito do Microcontrolador;



- Circuito LCD;

