

David Lohan Cardoso

**Sistema de controle de dispositivos para auxiliar
portadores de deficiência física e/ou visual**

Manaus - Amazonas

Janeiro - 2017

David Lohan Cardoso

Sistema de controle de dispositivos para auxiliar portadores de deficiência física e/ou visual

Modelo de Trabalho de Conclusão de Curso submetida à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro, em conformidade com as normas ABNT.

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST
Engenharia de Controle e Automação

Orientador Prof. Dr. Daniel Guzmán Del Rio
Coorientador: Prof. Mrs. Moisés Pereira Bastos

Manaus - Amazonas
Janeiro - 2017

David Lohan Cardoso

Sistema de controle de dispositivos para auxiliar portadores de deficiência física e/ou visual

Modelo de Trabalho de Conclusão de Curso submetida à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro, em conformidade com as normas ABNT.

Aprovado em 09 de Janeiro de 2017:

BANCA EXAMINADORA

Prof. Dr. Daniel Guzmán Del Rio
Orientador

Prof. M.Sc Moisés Pereira Bastos
Coorientador

Prof. M.Sc. Charles Luiz Silva de Melo
Presidente

Prof. André Dias de Lima Machado
Membro

Manaus - Amazonas
Janeiro - 2017

David Lohan Cardoso

Sistema de controle de dispositivos para auxiliar portadores de deficiência física e/ou visual/ David Lohan Cardoso. – Manaus - Amazonas, Janeiro - 2017
72 p.

Orientador Prof. Dr. Daniel Guzmán Del Rio

Trabalho Acadêmico – Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST
Engenharia de Controle e Automação, Janeiro - 2017.

1. Tecnologia Assistiva. 2. Portadores de deficiência física e/ou visual. 2. Sistema de controle de dispositivos. I. Daniel Guzmán Del Rio. II. Universidade do Estado do Amazonas. III. Escola Superior de Tecnologia. IV. Sistema de controle de dispositivos para auxiliar portadores de deficiência física e/ou visual

Aos meus pais, meus irmãos
e professores pelo incentivo
para realização deste
trabalho.

Agradecimentos

Agradeço primeiramente a Deus por ter me guiado nesta difícil jornada. Agradeço a minha mãe/vó Eliete de Oliveira Cardoso por todo incentivo e apoio nos momentos mais complicados, que não foram poucos, nesta caminhada. A minha mãe Simone Cardoso Silva que ajudou-me a chegar até onde estou. Agradeço principalmente ao meu pai Sérgio de Oliveira Cardoso que sempre dedicou seu tempo a mim, auxiliando em todas áreas da minha vida, incluindo os diversos conselhos que foram primordiais para eu ser o homem que sou e nos momentos mais difíceis, onde sempre esteve ao meu lado. A minha tia Solange de Oliveira Cardoso e meu tio Dirley de Oliveira Cardoso pelo incentivo diário.

A todos os meus amigos de graduação que ajudaram de alguma forma a chegar até esta etapa final do curso, principalmente meu amigo Josué Caldas que está ao meu lado desde o início da faculdade, auxiliando muito neste projeto, agradeço também meus amigos Nedom Pereira e Manoel Neto.

Ao meu orientador Daniel Guzmán Del Rio e meu coorientador Moisés Pereira Bastos, os quais foram muito importantes para a realização deste projeto.

Agradeço também meu coordenador Charles Luiz Silva de Melo que sempre apoiou a todos do curso de engenharia de controle e automação de maneira igualitária, ajudando no possível para termos uma formação acadêmica com excelência.

“O período de maior ganho em conhecimento e experiência é o período mais difícil da vida de alguém.”
(Dalai Lama)

Resumo

O desenvolvimento deste projeto é focado na Tecnologia Assistiva, área inovadora que utiliza a tecnologia atual para ajudar pessoas com diversas deficiências, sendo empregado em vários trabalhos por interagir com a sociedade. O presente projeto visa auxiliar os portadores de deficiência física e/ou visual nas atividades diárias ligadas aos dispositivos da residência, modificando o meio de controle físico por voz, deste modo, facilita a vida dos usuários, reduzindo a diferença social existente entre os portadores de deficiência física e/ou visual e as pessoas sem deficiência neste quesito. O sistema de controle de dispositivos é montado e acoplado em um recipiente feito em compensado organizando os componentes e sendo facilmente colocado em qualquer local, os dispositivos conectados e comandados por voz neste projeto são: televisão, ar condicionado, som, fechadura eletromecânica e a lâmpada. O projeto é desenvolvido e mostrado por etapas, composta de capítulos, seções e subseções até a interação final entre usuários e os dispositivos. O custo do projeto é altamente relevante para que qualquer pessoa possa comprar os materiais necessários e repita os procedimentos do trabalho.

Palavras-chaves: Tecnologia Assistiva, Portadores de deficiência física e/ou visual, Sistema de controle de dispositivos, Diferença social.

Abstract

The development of this project is focused in Assistive Technology, innovation field that uses current technology to support people with many types of disabilities, by applying into many works for interacting with society. The present project aiming to support people with physical and/or visual disability on daily activities connected into a home device, modifying by voice control, in this way, facilitating users life, reducing social differences existents between people with physical and/or visual disability and people without in this issue. The device control system is assembled and plugged in a recipient cartonwood made and easily placed anywhere, the devices connected by voice control in this project are: TV, stereo, a/c, eletromecanic door lock and bulb. The project is developed showing steps composed by chapters, sections and subsections until the final interaction between users and devices. Project final cost is highly relevant to any person that can buy necessary materials and repeat this work procedures.

Key-words: Assistive Technology, People with physical and/or visual disability, Device controlle system, Social difference.

Lista de ilustrações

Figura 1 – Transistores	20
Figura 2 – Dispositivos controlados por meio do host	21
Figura 3 – Cores refletidas pelo prisma	22
Figura 4 – Receptores infravermelho	23
Figura 5 – Emissores infravermelho	24
Figura 6 – Funcionamento interno do relé	25
Figura 7 – Modulo rele	26
Figura 8 – Funcionamento interno do motor de passo	26
Figura 9 – Motores de passo	27
Figura 10 – Circuito interno do ULN2003	28
Figura 11 – Driver uln2003	28
Figura 12 – Arduinos	30
Figura 13 – Shields	31
Figura 14 – Gerador hidrelétrico	32
Figura 15 – Peça de plastico dentada acoplada ao motor de passo	35
Figura 16 – Peça secundaria montada para trabalhar com o motor de passo	35
Figura 17 – Peças de compensado para a montagem do protótipo	36
Figura 18 – Protótipo pronto e aberto frontalmente	37
Figura 19 – Protótipo pronto e fechado frontalmente	37
Figura 20 – Ferrolho utilizado com o protótipo	38
Figura 21 – Protótipo pronto testado com o ferrolho	38
Figura 22 – EasyVR 3.0	40
Figura 23 – Controles remotos	41
Figura 24 – Circuito decodificador de comandos enviados por infravermelho	43
Figura 25 – Circuito controlador de dispositivos pelo LED emissor infravermelho no proteus	44
Figura 26 – Circuito controlador de dispositivos pelo LED emissor infravermelho	45
Figura 27 – Tensão e corrente do circuito controlador de dispositivos pelo LED emissor infravermelho	45
Figura 28 – Circuito acionador da lâmpada no proteus desativado	46
Figura 29 – Circuito acionador da lâmpada no proteus ativado	47
Figura 30 – Circuito acionador da lampada	47
Figura 31 – Tensão e corrente do circuito acionador da lampada	48
Figura 32 – Circuito controlador da fechadura eletromecânica no proteus	49
Figura 33 – Circuito controlador da fechadura eletromecânica	49
Figura 34 – Tensão e corrente do circuito controlador da fechadura eletromecânica	50

Figura 35 – Lâmpada e fechadura conectados no arduino com o modulo de reconhecimento de voz	50
Figura 36 – Lâmpada, fechadura e LED emissor infravermelho conectados no arduino com o modulo de reconhecimento de voz	51
Figura 37 – Peças retiradas do compensado para montagem do protótipo	52
Figura 38 – Protótipo pronto aberto	53
Figura 39 – Protótipo pronto fechado	53

Lista de tabelas

Tabela 1 – Códigos e componentes	30
Tabela 2 – Dispositivos, quantidade de comandos utilizados dos 32 possíveis e comandos dados	39
Tabela 3 – Controles remotos e quantidade de comandos	41
Tabela 4 – Custo total dos objetos utilizados no sistema	56

Lista de abreviaturas e siglas

GND	<i>Graduated Neutral Density</i> ou Terra
NA	Normalmente Aberto
NF	Normalmente Fechado
CC	Corrente Contínua
CA	Corrente Alternada
CNC	Comando Numérico Computadorizado
CPU	<i>Central Processing Unit</i>
PWM	<i>Pulse-Width Modulation</i>
DVD	<i>Digital Versatile Disc</i>

Sumário

	Lista de ilustrações	9
	Lista de tabelas	11
	Sumário	13
1	INTRODUÇÃO	15
1.1	Problemática	15
1.2	Motivação	16
1.3	Justificativa	16
1.4	Objetivos	17
1.4.1	Objetivo Geral	17
1.4.2	Objetivo Específico	17
1.5	Metodologia	17
1.6	Estrutura do trabalho	18
2	REFERENCIAL TEÓRICO	20
2.1	Evolução da automação residencial	20
2.2	Breve história da luz infravermelha	21
2.3	Controle de dispositivos	22
2.3.1	Receptor infravermelho	23
2.3.2	Led emissor infravermelho	24
2.3.3	Modulo relé	24
2.3.4	Motor de passo	26
2.3.5	Driver ULN2003	27
2.4	Microcontroladores	29
2.4.1	Arduino	29
2.4.2	Bibliotecas	30
2.4.3	<i>Shields</i>	31
2.5	Dispositivos eletromecânicos	31
2.5.1	Fechadura eletromecânica	32
2.6	Proteus	32
3	MATERIAIS E MÉTODOS	34
3.1	Lei de ohm	34
3.2	Montagem da fechadura eletromecânica	34
3.3	Modulo de reconhecimento de voz	38

3.4	Identificação de dispositivos controlados por infravermelho	40
3.5	Sistema de controle de dispositivos	41
3.5.1	Decodificação de comandos infravermelho	42
3.5.2	Circuito decodificador de comandos enviados por infravermelho	42
3.5.3	Controle de dispositivos pelo LED emissor infravermelho	43
3.5.4	Circuito controlador de dispositivos pelo LED emissor infravermelho	44
3.5.5	Controle da lâmpada	45
3.5.6	Circuito acionador da lâmpada	46
3.5.7	Controle da fechadura eletromecânica	48
3.5.8	Circuito controlador da fechadura eletromecânica	48
3.6	Unificação dos circuitos de controle	50
3.7	Protótipo para armazenamento da central de controle e dos com- ponentes	51
3.8	Teste prático	53
4	CONSIDERAÇÕES FINAIS	55
4.1	Custo final do projeto	56
4.2	Trabalhos futuros	56
4.3	Trabalhos relacionados	57
A	APÊNDICE A - ALGORITMO RECEPTOR INFRAVERMELHO . .	58
B	APÊNDICE B - ALGORITMO TESTE 1	59
C	APÊNDICE C - ALGORITMO TESTE 2	60
D	APÊNDICE D - ALGORITMO FINAL	61
	REFERÊNCIAS	70

1 Introdução

O sistema de automação residencial consiste em compartilhar informações entre usuário e dispositivo, podendo ser de formas distintas, mas buscando facilidade na comunicação e no controle a ser realizado. Em certos casos, há dispositivos inclusos no sistema automação residenciais comandados por controles remotos, onde consiste basicamente na emissão e recepção de ondas infravermelhas, como televisão, rádio, ar-condicionado etc.

Os dados do Instituto Brasileiro de Geografia e Estatística (IBGE) mostram que 6,2% da população brasileira possui algum tipo de deficiência, sendo a visual mais representativa com 3,6% dos pesquisados, atingindo comumente idosos com mais de 60 anos (11,5%) e 16% dos deficientes visuais são incapazes de realizarem atividades habituais como ir à escola, trabalhar, etc. A deficiência física aparece em 1,3% dos pesquisados e quase metade (46,8%) tem um grau intenso ou muito intenso (IBGE, 2010).

Há inúmeras dificuldades diárias enfrentadas pelos portadores de deficiência física e/ou visual, isto transforma ações simples em complexas e normalmente necessita de auxílio para realizações de tarefas rotineiras. A maioria dos dispositivos é incapaz de interagir com os portadores de deficiência física e/ou visual, expandindo a desigualdade social existente. A desigualdade social na perspectiva da deficiência é caracterizada pelas diferenças sociais geradas na acessibilidade a determinados recursos, empregos e qualificações, enquadrando no âmbito político e social, etc. (PIMENTA; SALVADO, 2010).

O novo termo utilizado para identificar todos os itens, equipamentos, produtos e serviços que melhorem ou mantenham as capacidades funcionais dos portadores de deficiência é designado tecnologia assistiva e conseqüentemente favorecendo independência e inclusão (SARTORETTO; BERSCH, 2014). O sistema de automação residencial pode ser utilizado para facilitar e, todavia melhorar a vida dos portadores de deficiência, pois oferece o controle alternativo dos dispositivos agregados viabilizando superioridade no desempenho.

Buscando associar os problemas diários enfrentados pelos portadores de deficiência física e/ou visual, tais como ligar ou desligar a televisão, modificar a temperatura do ar-condicionado, alterar a estação do rádio etc. com o sistema de automação residencial, fora elaborado o presente projeto que os ajudará de forma que os comandos aos dispositivos serão dados por voz.

1.1 Problemática

A maioria das pessoas com deficiência tem suas habilidades funcionais reduzidas, logo, algumas atividades são dificultadas. Simples ações como acessar um dispositivo na residência para as pessoas com deficiência física e/ou visual se tornam complexas, pois, a mobilidade e/ou a visibilidade para ter o controle eficiente do dispositivo é ineficaz. A

diferença entre pessoas sem e com deficiência física e/ou visual é grande em relação à acessibilidade dos dispositivos e muitos dispõem de um sistema para auxiliar o controle, tais como, comandos por voz, conexão *bluetooth* ou *Wi-Fi*, etc.

O avanço da tecnologia gerou e ainda gera diversos sistemas para auxiliar pessoas com deficiência, porém, não são acessíveis a todos. A implementação de um sistema para se comunicar com os usuários, por exemplo, por voz, é mais sofisticado e requer um custo maior, por isto, muitos portadores de diversas deficiências, principalmente física e/ou visual, tem dificuldades para utilizar alguns dispositivos. Os dispositivos que possuem um sistema para realizar comunicação por voz com os usuários, na maioria das vezes, são utilizados para comodidade conforme a equação.

1.2 Motivação

Buscando formas alternativas para auxiliar as pessoas com deficiência física e/ou visual nas dificuldades enfrentadas em suas atividades diárias, diversas propostas foram dadas para tais problemas, porém, sempre priorizando a forma mais eficiente e com menor o custo, dando assim a oportunidade de ter uma vida normal em relação ao controle de alguns dispositivos na residência.

O projeto em questão será um sistema que enviará comandos aos dispositivos por voz, facilitando o controle e oferecendo maior comodidade. A comunicação por voz melhorará o uso do dispositivo para os portadores de deficiência física e/ou visual por possibilitar o envio de comandos em qualquer parte dentro do raio máximo alcançado pelo sistema, sem a necessidade da locomoção até o dispositivo ou utilizar o controle remoto.

O autor do projeto procurou formas de ajudar pessoas com deficiências relacionando o conhecimento adquirido e as dificuldades enfrentadas, criando assim este trabalho.

1.3 Justificativa

Há inúmeras dificuldades no dia-a-dia dos portadores de deficiência física e/ou visual, dependendo do grau, os obstáculos se tornam maiores. Muitas vezes é necessário auxiliar os portadores de deficiência física e/ou visual nas atividades diárias sendo, às vezes, inconveniente para quem ajuda e constrangedor para o portador da deficiência. A necessidade do ajudante para os portadores de deficiência física e/ou visual, em certos casos, é inevitável, e o desejo pela independência social é cada vez maior.

Deste modo, este projeto é justificado pela facilidade em controlar dispositivos, fornecendo maior independência social aos portadores de deficiência física e/ou visual. Nesse sistema, os dispositivos são controlados por voz, todavia, é mais acessível para os portadores de deficiência física e/ou visual, pois não é necessário o auxílio de pessoas para realizar atividades diárias como ligar ou desligar a televisão, o rádio, o ar-condicionado,

etc. É possível encontrar dispositivos que fazem a interação com os usuários por comandos de voz, porém, são tecnologicamente avançados e, todavia, caros.

Este projeto é uma maneira simples e prática de solucionar algumas dificuldades cotidianas dos portadores de deficiência física e/ou visual com o custo financeiro baixo comparando a compra de um dispositivo com esta função.

1.4 Objetivos

1.4.1 Objetivo Geral

Desenvolver um sistema, onde seja possível controlar dispositivos por comandos dados por voz para auxiliar portadores de deficiência física e/ou visual. Este projeto visa facilitar a comunicação entre diversos dispositivos e os portadores de deficiência física e/ou visual fornecendo maior independência.

1.4.2 Objetivo Específico

1. Realizar o enredo bibliográfico do tema.
2. Obter a leitura dos comandos enviados pelos controles remotos dos dispositivos.
3. Criar o protótipo de controle da fechadura eletromecânica
4. Montar os circuitos para o controle de dispositivos, acionador da lâmpada e controlador da fechadura eletromecânica
5. Desenvolver o algoritmo para funcionamento em conjunto do circuitos controladores dos dispositivos.
6. Criar o protótipo para o armazenamento da central de controle e dos componentes
7. Unir os circuitos juntamente com os componentes necessários.
8. Testes e ajustes do sistema.

1.5 Metodologia

Inicialmente será realizado o levantamento bibliográfico sobre o tema, buscando materiais e métodos em projetos semelhantes para aplicações no sistema criado nesta monografia.

O sistema do projeto apresentado nesta monografia requer o uso de componentes e de placas de controle para o funcionamento eficaz, tais como: LED emissor infravermelho,

LED receptor infravermelho, transistor, rele, microcontrolador e *shield* de reconhecimento de voz, etc.

O projeto em questão auxiliará os portadores de deficiência física e/ou visual no controle de dispositivos por voz, logo é necessário estudar como os comandos são normalmente enviados e recebidos, antes da montagem do sistema, para realizar as modificações e alterar o modo de controle, facilitando a uso do dispositivo.

Os dispositivos controlados por controle remoto terão alguns comandos lidos por um LED receptor infravermelho conectado ao microcontrolador e utilizando a programação necessária. Os códigos emitidos serão devidamente anotados para futuramente utilizar no controle do dispositivo com o LED emissor infravermelho.

O controle dos dispositivos será realizado por voz logo, os comandos devem ser gravados no *shield* de reconhecimento de voz e quando acionados, ativar os componentes do sistema realizando o controle.

A montagem do sistema será o próximo passo, juntando todos os componentes e fazendo a ligação física com os dispositivos que não podem ser controlados pelo LED emissor infravermelho.

O desenvolvimento da programação será realizado e depois de finalizado será compilado no microcontrolador, sendo este à base de controle do sistema, fazendo com que os componentes e o *shield* funcionem em conjunto.

Enfim, dará início a etapa final do projeto, onde serão realizados os testes e será verificado se o sistema está funcionando de acordo com o estipulado.

1.6 Estrutura do trabalho

Este trabalho está dividido em quatro capítulos, apresentados da seguinte forma:

- Introdução (Capítulo I)
- Referencial Teorico (Capítulo II)
- Materiais e Métodos (Capítulo III);
- Resultados e Discussão (Capítulo IV);

No Capítulo 1 é apresentado inicialmente uma breve introdução sobre o tema auxiliado de pesquisas reais ligadas a pessoas com deficiência física e/ou visual. Na continuação do capítulo são mostrados os sequencialmente os problemas a serem seguidos, o motivo da escolha do tema, a justificativa, o objetivo geral e específico e a metodologia seguida.

O Capítulo 2 mostra a historia da evolução da automação sendo este o foco do projeto ao final do desenvolvimento. Sequencialmente mostra a historia da descoberta do infravermelho, sendo essencial na utilização de alguns dispositivos e são apresentados os

componentes usados no projeto de uma maneira geral, definindo as funções dando exemplos de cada.

O Capítulo 3 é o principal, pois, neste é abordado o desenvolvimento do projeto por etapas como: definição dos modelos de componentes apresentados no capítulo II, fases das montagens dos protótipos, conexões dos componentes, entre outros. Neste capítulo é montado o sistema geral de controle de dispositivos.

No Capítulo 4 é realizada a discussão sobre os resultados do sistema, verificando se o objetivo fora alcançado.

2 Referencial Teórico

Para a escrita deste capítulo fora necessário realizar o enredo bibliográfico sobre o tema, buscando componentes e placas eletrônicas que se comuniquem com os dispositivos.

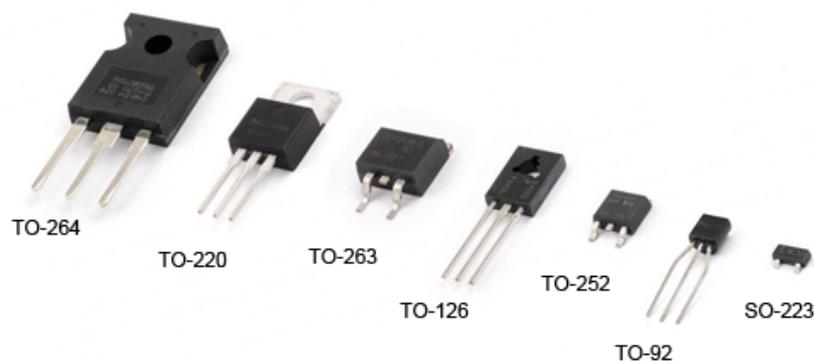
Os portadores de deficiência física e/ou visual se comunicam com os dispositivos através de comandos de voz. Este capítulo engloba todos os componentes de modo geral, exemplificando diversos modelos reais. As placas eletrônicas de reconhecimento de voz e a de comunicação com os dispositivos são primordiais para o desenvolvimento do sistema e para realizar a comunicação com os portadores de deficiência, sendo estas, mostradas neste capítulo.

O desenvolvimento do projeto é realizado pelos componentes e as placas eletrônicas abordados neste capítulo.

2.1 Evolução da automação residencial

A invenção que mudou o mundo tecnológico, diminuindo o tamanho dos dispositivos e aumentando sua capacidade significativamente fora inventado em meados de 1945 dando o premio nobel de física para seus criadores em 1956 (John Bardeen, Walter H. Brattain e William Shockley) (BASALLA, 2011). Com a criação dos transistores, diversos dispositivos tomaram novas formas e funções, sendo muito utilizado até nos dias atuais, em microprocessadores de computadores, por exemplo, há milhões em um único *chip*. Os computadores são de grande importância para a evolução da automação residencial pela interface de controle entre usuários e dispositivos em alguns casos, e pela comunicação com os microcontroladores, utilizados como centro de controle do sistema. Exemplos de transistores são ilustrados na figura 1.

Figura 1 – Transistores



Fonte: luisllamas.es/wp-content/uploads/2016/06/arduino-transistor-bjt-componente.png

A comunicação sem fio iniciou em 1888 por Heinrich Rudolf, produzindo as primeiras ondas de radio, nos anos seguintes foram aprimoradas e eram consideradas uma forma de comunicação. Em 1899 até 1901 Guglielmo Marconi Marchese amplificou as ondas fazendo com que viajassem por milhas, chegando ao ápice da distância quando atravessou o oceano atlântico (LUIZ, 2013). A evolução desta comunicação é de extrema importância para a automação residencial, pois, é possível realizar diversas atividades na residência por equipamentos que trabalham sem fio. A sociedade é modificada juntamente com a evolução tecnológica aumentando o comodismo oferecido pelos diversos dispositivos.

A evolução dos dispositivos e componentes eletrônicos proporciona maior comodidade e segurança para a sociedade, sendo, comumente utilizado em residências. Muitos dispositivos possuem funções alternativas de controles, tais como: comandos por voz, *bluetooth*, *Wi-Fi*, podendo ser utilizado pelo celular e outros aparelhos que interajam, em qualquer parte mediante o alcance máximo do dispositivo, sendo mais cômodo para o usuário pela praticidade e facilidade oferecida todavia, pode ser utilizado para auxiliar os portadores de deficiências em atividades diárias. É possível controlar automaticamente ou na presença de pessoas várias partes da residência ou qualquer outro lugar da escolha do usuário, incluindo fechaduras, televisão, som, câmeras, lâmpadas, ar-condicionado, entre outros. Na figura 2 é possível visualizar diversos dispositivos em vários cômodos da residência sendo controlado por apenas um eletrônico.

Figura 2 – Dispositivos controlados por meio do host



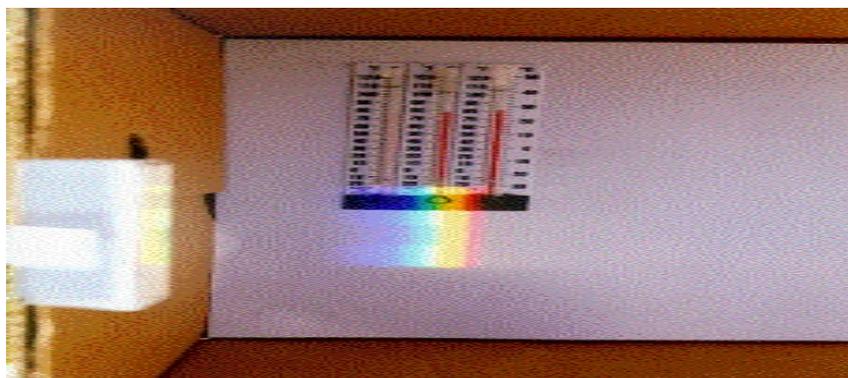
Fonte: benx.com.br/blog/wp-content/uploads/2016/03/automacao-maquete.jpg

2.2 Breve história da luz infravermelha

Em 1800 fora descoberta a existência do infravermelho por Sir Willian Herschel no experimento onde fez a luz solar atravessar um prisma, deste modo, diferentes cores foram mostradas. As cores refletidas pelo prisma são as mesmas do arco-íris, variando do tom azul ao vermelho, com diferentes temperaturas. Herschel posicionou um termômetro na cor azul e percebeu que a temperatura crescia até o tom vermelho (esta é a região visível), quando a medida fora realizada além do tom vermelho, verificou que era mais alta, conclui

a existência de uma luz que não podia ser vista, nomeada infravermelho. Na figura 3 mostra as diversas cores refletidas pelo prisma e as respectivas temperaturas (mesmo das cores invisíveis para o ser humano).

Figura 3 – Cores refletidas pelo prisma



Fonte: [if.ufrgs.br/rif-](http://if.ufrgs.br/rif-fel/notas_aula/ensino_astro/roteiros/Roteiro_Experimento_Herschel_files/image005.gif)

[fel/notas_aula/ensino_astro/roteiros/Roteiro_Experimento_Herschel_files/image005.gif](http://if.ufrgs.br/rif-fel/notas_aula/ensino_astro/roteiros/Roteiro_Experimento_Herschel_files/image005.gif)

Há relatos que os primeiros controles com infravermelho foram utilizados na primeira guerra mundial para controlar pequenos navios. Na segunda guerra mundial utilizaram os controles para detonar bombas. Com o fim das guerras a tecnologia utilizada juntamente com o infravermelho fora deixada de lado por anos, até então ser novamente lembrada no controle das TVs (JOBSTRAIBIZER, 2010).

O controle remoto funciona emitindo códigos binários aos dispositivos (0 e 1), criando sequências e cada sendo certa função como desligar ou liga a TV, aumentar o volume, alterar o canal, etc. Há um LED emissor infravermelho nos controles, no qual se comunica com os dispositivos, as vezes é necessário posicionar frontalmente com o dispositivo para realizar o envio dos comandos, pois o LED é potencialmente fraco (JOBSTRAIBIZER, 2010).

O infravermelho fora a descoberta que ofereceu maior comodidade (mesmo depois de muito tempo) a população, com a criação do controle remoto e outros dispositivos que usufruem desta brilhante tecnologia. É possível enviar diversos comandos em qualquer parte dentro do raio de comunicação entre o controle remoto e o dispositivo, sentado na cadeira, deitado na cama, ou seja, podendo ser distante do dispositivo.

2.3 Controle de dispositivos

O controle dos dispositivos será realizado com o auxílio de componentes como LED emissor e receptor infravermelho.

Os dispositivos que não são comandados pelos LEDs emissores serão conectados diretamente ao sistema para ter o controle.

Alguns componentes como *driver* e relé serão utilizados juntamente com o arduino, conectados diretamente com a iluminação do local e com o motor de passo que realizará o controle do trancamento e abertura da porta.

Todos os comandos serão emitidos por voz gerando a atuação individual em cada dispositivo de acordo com o reconhecimento dos comandos gravados.

2.3.1 Receptor infravermelho

São componentes que captam a luz infravermelha no ambiente, onde podem ser usados em diversas utilidades, exemplos: sensores de presença (verifica se há radiação infravermelha decorrente do calor emitido pelos corpos), leitura de pulsos infravermelhos enviados por controle remoto, entre outras.

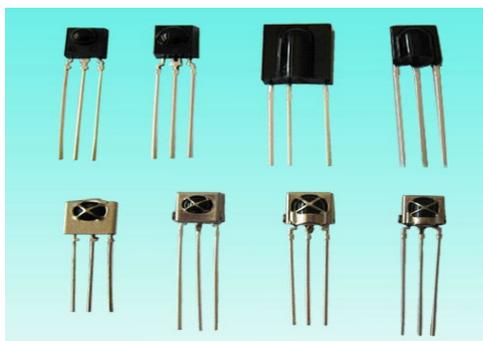
A frequência de trabalho do componente é de extrema importância, caso queira decodificar os códigos enviados por um emissor infravermelho, se forem diferentes, a decodificação não será realizada corretamente.

Os Receptores Infravermelhos mais comuns trabalham com a frequência de 38KHz (maioria dos controles remotos) e possuem três pinos que analisados frontalmente, da esquerda para a direita, são respectivamente: *Out* (Sinal), GND (Terra) e *Vcc* (alimentação interna do receptor (normalmente 5 volts). Esta sequência pode ser alterada dependendo do modelo (LOMBARDI, 2006).

Alguns modelos dos receptores infravermelhos com três pinos são: TSOP1838, SFH505A, PIC12043S, etc.

Existem receptores que possuem dois pinos, porém, não podem ser utilizados para decodificar sinais infravermelhos, apenas para verificar a incidência de luz infravermelha no ambiente, são chamados de fototransistores. Fisicamente parecidos com um LED e funcionalmente iguais a um transistor, a luz infravermelha faz o papel de base, ou seja, atua como uma chave, liberando ou não a corrente entre coletor e emissor, exemplos: TIL78, TIL67, TIL66, etc. Na figura 4 são ilustrados diversos receptores infravermelho.

Figura 4 – Receptores infravermelho



Fonte: <https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT-7F1ydjn9UexoB-vV-w5XdGXevilBUW-w2PWcHw3qOtQe3Aev>

2.3.2 Led emissor infravermelho

São componentes que emitem luz infravermelha continuamente ou em forma de pulsos, possui dois pinos de diferentes tamanhos, sendo estes: ânodo (maior) e cátodo (menor), positivo e negativo, respectivamente. A corrente transitória do emissor infravermelho flui em um único sentido, igualmente ocorre nos diodos. Sua aplicação é visível em: Controles Remotos, Sensores de objetos, etc. Dependendo do modelo a capsula tem o tamanho variável (normalmente entre três a dez milímetros) elevando o alcance máximo e a corrente, tensão e frequência suportáveis são diferentes, exemplos: TIL32, TIL38, TIL34, etc.

A combinação deste componente com o receptor infravermelho é muito importante quando se deseja controlar dispositivos que possuem controles remotos. Inicialmente, verifica-se a frequência dos componentes e do controle remoto, pois, devem ser iguais para que o código do comando não seja diferente ou inexistente. É possível utilizar os códigos obtidos em um LED emissor infravermelho, todavia os comandos (CAVALCANTE; RODRIGUES; BUENO, 2013). Este processo de emissão-recepção da luz infravermelha e obtenção dos códigos pode ser realizado com o auxílio de um microcontrolador. Exemplos de emissores infravermelhos são mostrados na figura 5.

Figura 5 – Emissores infravermelho



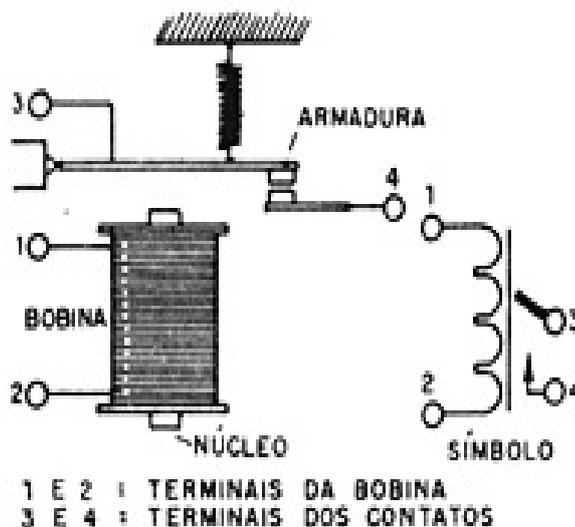
Fonte : <http://image.made-in-china.com/2f0j10RZYEGOpJbiqS/-Diodo-emissor-de-luz-infravermelho-da-tev-.jpg>

2.3.3 Modulo relé

O relé é um componente eletromecânico, internamente possui a armadura, mola, eletroímã e alguns contatos acionados eletricamente. Inicialmente, quando não há transição de corrente na bobina, o contato elétrico da armadura está no pino NF. A transição de

corrente na bobina gera um campo magnético que atrai o contato elétrico da armadura, conectando outros dois terminais ativando o circuito nesta conexão, agora o estado do pino muda de NF para NA. Quando a corrente é interrompida, a mola faz com que a armadura volte para o estado inicial. O relé é utilizado para isolar dois circuitos com segurança (EDITORA, 2015). A figura 6 ilustra o funcionamento interno do relé.

Figura 6 – Funcionamento interno do relé



Fonte : <http://www.newtoncbraga.com.br/images/stories/artigos2/fig1.gif>

Há diversas utilidades para o relé tais como acender uma lâmpada, acionar um motor, setas ou até o sistema de água dos para-brisas do carro, sendo estes apenas alguns exemplos. O relé está presente em quase todos os sistemas eletrônicos, são encontrados nos ar-condicionados, elevadores, carros, etc.

É possível encontrar vários tipos de relés, tais como relé de impulso, relé fotoelétrico e o relé simples, facilmente encontrados em muitos sistemas eletrônicos. O relé fotoelétrico é fundamentalmente utilizado para acionar lâmpadas e equipamentos elétricos, funcionando com a ação de uma fonte luminosa. Este componente é notoriamente utilizado em sistemas de iluminação pública. O relé de impulso auxilia no controle automático de vários equipamentos e no sistema de iluminação. Tem a funcionalidade interna parecida com a do relé simples, onde, quando recebe um pulso de tensão na bobina, modifica o contato, porém, oferece diversas vantagens ao utilizar (EDITORA, 2015).

O módulo relé é um componente para facilitar a comunicação entre os dispositivos e a central de comando, muitas vezes o microcontrolador faz essa tarefa. O microcontrolador trabalha com tensões e correntes baixas, podendo danificar o sistema caso ultrapasse o valor máximo especificado. O módulo relé é largamente utilizado para controlar um ou mais dispositivos juntamente com o microcontrolador enviando baixas tensões.

Há módulos com dois ou mais relés conectados a placa, sendo usado para vários dispositivos com fácil comunicação e manipulação para o usuário. É possível controlar dispositi-

tivos com tensões altas como 110V ou 220V utilizando uma baixa tensão para energizar a bobina como 6V ou 12V (BRAGA, 2010). Na figura 7 é mostrado o modelo de modulo relé comumente utilizado com o microcontrolador, com disponibilidade de controlar um dispositivo.

Figura 7 – Modulo relé

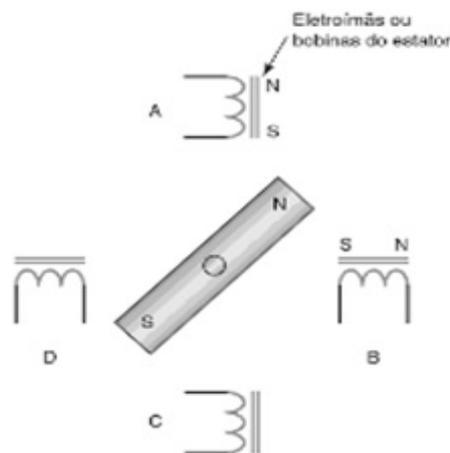


Fonte: static.mercadoshops.com/modulo-rele-5v-para-arduino-arm-avr-dsp-pic-pronta-entrega-iZ6XvZxXpZ4XfZ152471996-535366080-4.jpgXsZ152471996xIM.jpg

2.3.4 Motor de passo

É um motor CC que não gira suavemente e continuamente igual aos outros motores, mas em passos ou incrementos. O motor de passo é composto por eletroímãs (bobinas de estator) e o rotor, que é um ímã permanente. Ao energizar as bobinas o rotor move, pois cria um ímã que atrai a barra do rotor. Apenas uma bobina está em estado *ON*, as outras estão *OFF*, ou seja, é possível gerar vários movimentos energizando e desenergizando as bobinas do motor (FRENZEL, 2015). Na figura 8 é mostrado o funcionamento interno do motor de passo com quatro eletroímãs ou bobinas.

Figura 8 – Funcionamento interno do motor de passo



Fonte : Livro eletrônica moderna, pagina 186

Há diferentes tipos de motores de passo tais como relutância variável de pilha simples e múltipla e os do tipo PM, ou seja, alguns com mais bobinas, aumentando a quantidade de passos por volta, sendo mais preciso em certas ocasiões. O número de passos que o motor pode dar é proporcional à quantidade de bobinas encontradas em seu interior, alternando entre *ON* e *OFF*, é possível ter um controle preciso da rotação (LAMB, 2015).

Algumas características do motor de passo são: inexistência de escovas, movimentação com cargas (desde que não exceda o torque máximo do motor), manter o eixo estacionário, ótima resposta a partida, parada e reversão de movimento.

O motor de passo tem um torque razoável, podendo ser utilizado para diversas atividades, juntamente com um microcontrolador ou outro dispositivo que realize o controle. É possível encontrar o motor de passo em dispositivos como impressoras a jato de tinta, CNCs e bombas volumétricas (CONDIT; JONES, 2004). Na figura 9 é mostrado diferentes modelos de motores de passo utilizados em diversos projetos.

Figura 9 – Motores de passo

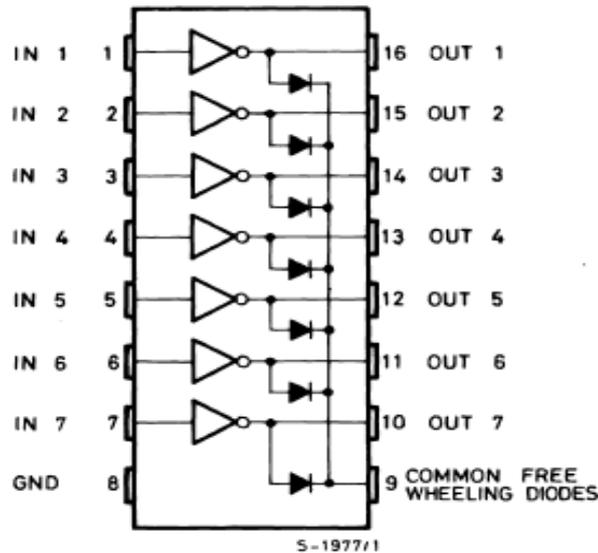


Fonte: <https://www.usinainfo.com.br/img/cms/fotos-categorias/motores-de-passo.jpg>

2.3.5 Driver ULN2003

O *driver* ULN2003 é formado por sete transistores darlington NPN e pode manipular até 500mA por saída. Este é um *driver* de alta tensão e corrente, possui resistor de 2,7k Ω para cada saída que pode ser conectada a diversos dispositivos compatíveis (RAMAMURTHY; BHARGAVI; SHASHIKUMAR, 2010). Na figura 10 é mostrado o circuito interno do *driver* ULN2003.

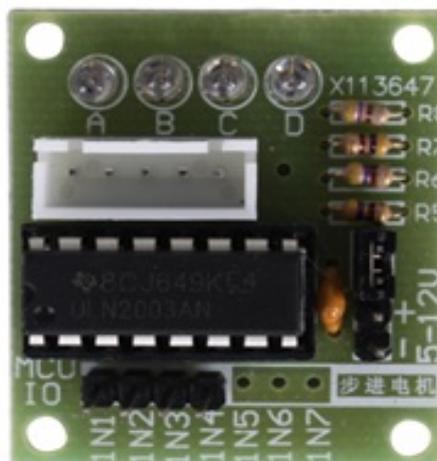
Figura 10 – Circuito interno do ULN2003



Fonte: ni.i.lithium.com/t5/image/serverpage/image-id/135284i0D5B1F993930EF03?v=v2

O *driver* ULN2003 é um componente utilizado juntamente com o arduino para controlar dispositivos que necessitem de uma corrente excedente. Este *driver* consiste em receber pequenas correntes enviando maiores em suas saídas, podendo controlar dispositivos como motores de passo, motor CC ou relés. A atuação deste *driver* é recomendada com motores de passo, pois, atua comumente com este dispositivo. Na figura 11 é mostrado o *driver* ULN2003 frontalmente, comumente acoplado com o motor de passo.

Figura 11 – Driver uln2003



Fonte: <https://bwgz57.files.wordpress.com/2012/04/2720-p-1320673805227.jpg>

2.4 Microcontroladores

O microcontrolador é composto por um microprocessador e outros componentes essenciais para seu funcionamento interligados em um único circuito. Consomem pouca energia e possuem um baixo custo (SANTOS, 2009).

São funcionalmente iguais a um computador, no qual inclui uma CPU, memória de dados e programa, portas de entrada e saída (I/O), sistema de *clock* e outros possíveis periféricos, como: temporizadores, conversor analógico digital, etc.

Alguns exemplos de microcontroladores são: ATmega, ARM, PIC, entre outros.

2.4.1 Arduino

O arduino é um pequeno computador programado para processar entradas ou saídas, podendo ser ambas, juntamente com componentes externos conectados a ele, é utilizado para realização de vários projetos em diferentes áreas. (MCROBERTS, 2011).

A corrente e tensão máxima que podem ser fornecidas é de respectivamente, 40mA e 5V, caso ultrapasse esses valores é possível danificar o arduino.

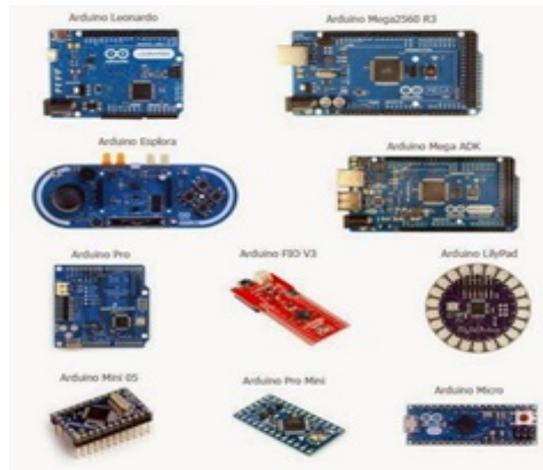
Sabe-se que o arduino tem seu microcontrolador embutido na placa para facilitar as ligações e utilização das portas, o ATmega, assim chamado, é onde ocorre as operações e comanda cada porta existente na placa.

Existem vários modelos do arduino, deste modo varia o numero de portas, tamanho da placa, etc. Exemplos de modelos: Arduino Uno, Arduino Yún, Arduino Nano, entre outros. É possível encontrar três diferentes tipos de portas nos arduinos, são estas:

- Portas digitais: Assumem dois estados (ligada ou desligada), enviando a tensão máxima possível (normalmente 5V) quando ativa, caso contrario, a mínima (0V)
- Portas PWM: São portas que podem variar a tensão fornecida ao componente ou dispositivo, podendo assumir valores entre 0V a 5V.
- Portas Analógicas: São normalmente entradas, utilizadas para leitura de sensores, podendo assumir valores entre 0V a 5V como resposta. O microcontrolador do arduino trabalha apenas com valores digitais então é necessário traduzir os valores recebidos, existe um conversor A/D que faz esse procedimento.

Diferentes modelos reais de arduinos são mostrados na figura 12.

Figura 12 – Arduinos



Fonte: 2.bp.blogspot.com/-4vSJEOW6JE/VBIw2luMp2I/AAAAAAAAAfQ/aHDkHTKN92w/s1600/arduino2.jpg

2.4.2 Bibliotecas

Bibliotecas são extensões para facilitar a comunicação e a programação com hardwares externos e o arduino. Englobam uma serie de variáveis quando utilizadas, diminuindo o tamanho do algoritmo.

Algumas bibliotecas são instaladas juntamente com a plataforma de programação do arduino, mas é possível baixar ou criar uma biblioteca, caso queira (ARDUINO, 2012a).

É necessário conhecer as bibliotecas prontas pelo arduino para utilizar o código e em qual componente funciona, alguns exemplos destas bibliotecas são mostrados na tabela 1.

Tabela 1 – Códigos e componentes

Código	Componente
<code>include <Servo.h></code>	Servomotores
<code>include <Wifi.h></code>	Arduino WiFi shield
<code>include <Ethernet.h></code>	Arduino Ethernet shield
<code>include <LiquidCrystal.h></code>	LCD
<code>include <Stepper.h></code>	Motores de passo

Fonte: Autor

Ao ativar o código do componente, vários comandos são liberados e devem ser utilizados de acordo com a sequência e metodologia da programação criada pelo usuário. Há exemplos de algoritmos prontos de alguns *shields* no site oficial do arduino.

2.4.3 Shields

São *hardwares* externos que podem ser conectados diretamente ao arduino expandindo sua capacidade e inserindo novas funcionalidades. Há vários tipos diferentes de *shields*, porém, seguem a mesma filosofia, sendo fácil de montar e barato para produzir (ARDUINO, 2012b).

Os *shields* são fáceis de manipular com o auxílio das bibliotecas. Exemplos de *shields*: Xbee, EasyVR, Ethernet, etc. São mostrados modelos reais de *shields* na figura 13 que são acoplados com arduino.

Figura 13 – Shields



Fonte: <http://bertoli.tech/wp-content/uploads/2016/09/List-Arduino-Shields-Thumbnail-1024x667.jpg>

2.5 Dispositivos eletromecânicos

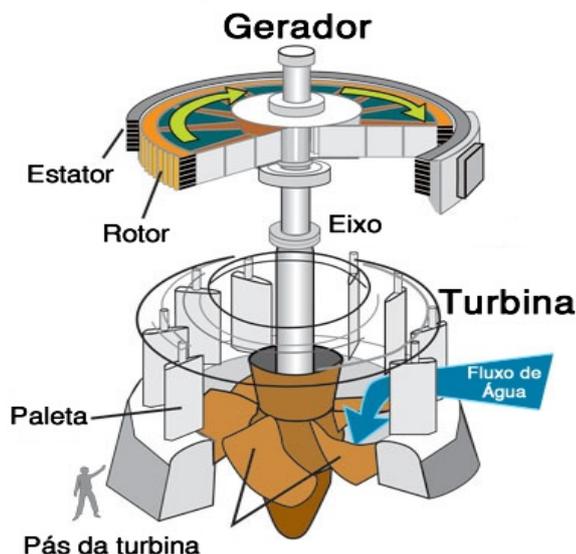
Com surgimento de tecnologias como a máquina a vapor, o motor, o dínamo, turbina hidráulica e a lâmpada nos séculos XVIII e XIX, apresentou a possibilidade na conversão de energia mecânica em elétrica, dando início a era dos dispositivos eletromecânicos (PEREIRA, 2015).

Os dispositivos eletromecânicos convertem energia elétrica em mecânica ou vice-versa (PINTO, 2011). Internamente nos dispositivos eletromecânicos, é possível encontrar componentes elétricos e mecânicos, não podem ser separados fisicamente, apenas pela conversão de energia.

Atualmente, é possível encontrar esta conversão de energia mecânica em elétrica nas usinas hidrelétricas. Há grandes máquinas rotativas geradoras e a passagem elevada de água pelas pás das turbinas gera um movimento mecânico, posteriormente convertido em

energia elétrica (BURATTINI, 2008). Na figura 14 é ilustrado o funcionamento de um gerador da hidrelétrica.

Figura 14 – Gerador hidrelétrico



Fonte: infoescola.com/wp-content/uploads/2008/09/hidroeletrica-turbina.jpg

Há outros dispositivos utilizados comumente em residências ou locais que atuam realizando esta conversão de energia elétrica em mecânica, ou vice-versa, tais como ventiladores, máquinas de lavar, secadores de cabelo, etc.

2.5.1 Fechadura eletromecânica

A Fechadura de acionamento eletromecânico é comandada eletricamente por um microcontrolador ou qualquer dispositivo que realize este controle. Enviando pulsos elétricos, internamente controlará o motor de passo que realiza a ativação ou desativação da fechadura, locomovendo a barra.

Há dispositivos no mercado que resultam na conversão de energia elétrica acionando mecanicamente equipamentos como fechaduras, todavia, são caros e dificilmente encontrados, deste modo, a construção de uma fechadura eletromecânica é mais viável.

O acionamento da fechadura eletromecânica é realizado através do algoritmo de controle, incluindo comandos enviados por voz. A fechadura eletromecânica é composta por um motor de passo, matérias como barras e pequenos pedaços em plásticos e compensados, entre outros.

2.6 Proteus

Proteus é um programa utilizado para montagem virtual de vários circuitos, tendo diversas funcionalidades como cálculo de corrente, tensão, resistência, em CC ou CA. Os

microcontroladores podem ser usados no desenvolvimento dos circuitos virtuais juntamente com o algoritmo (ZHANG, 2012).

Os componentes que compõem o sistema de controle de dispositivos por comandos vocais, serão montados e testados inicialmente no proteus.

O teste do sistema é feito por etapas iniciando com o LED receptor mostrado na subseção 2.3.1 utilizado para a leitura dos comandos enviados pelo controle remoto do dispositivo. A montagem virtual será realizada após a análise do *datasheet* do componente, buscando o valor da tensão e corrente informado.

Seguindo as montagens virtuais de cada componente para o teste do sistema real, será realizada a montagem do LED emissor encontrado na subseção 2.3.2. Após isto os dispositivos que não são controlados por infravermelho terão os circuitos montados no proteus utilizando os componentes citados nas subseções 2.3.3, 2.3.4 e 2.3.5.

É possível utilizar microcontroladores no proteus, deste modo, o circuito será conectado juntamente com o arduino citado na subseção 2.4.1 e a programação individual.

A montagem do circuito virtual é de extrema importância para o funcionamento eficaz do sistema real, pois, as chances de exceder os valores informados no *datasheet* diminuem, logo, reduz a possibilidade em queimar os componentes.

3 Materiais e métodos

Neste capítulo é desenvolvido o projeto seguido de etapas visando facilitar o entendimento e a montagem do sistema.

Os componentes anteriormente mostrados são especificados para a montagem do sistema, sendo testado em cada circuito que realiza a comunicação com um dispositivo específico.

Os circuitos são montados independentemente para que ao final do capítulo seja unido com todos e deste modo, desenvolva o sistema de controle de dispositivos final do projeto. Os circuitos são testados com os respectivos algoritmos.

A criação dos protótipos é demonstrada em varias etapas neste capitulo.

O teste final do sistema é realizado quando o projeto está montado, a verificação da atuação dos dispositivos é levada em consideração e os resultados são discutidos no capitulo posterior.

3.1 Lei de ohm

A lei de ohm fora apresentada em 1826 pelo físico Georg Simon Ohm, expressa a relação entre corrente, tensão e resistência, sendo as unidades respectivamente amperes, volts e ohm. Esta lei pode definir os valores de praticamente todos os circuitos CC (EISMIN, 2016). A corrente é diretamente proporcional à tensão e inversamente à resistência, logo a formula é expressa por:

$$U = R.I \quad (3.1)$$

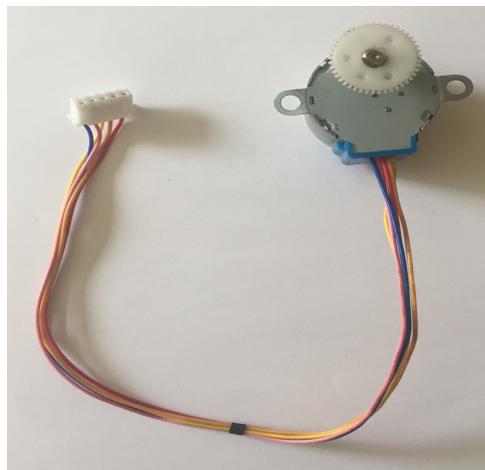
Esta equação é utilizada no projeto para verificar a corrente, tensão e resistência nos LEDs emissores e receptores apresentados nas subseções 2.3.2 e 2.3.1, respectivamente. Deste modo é possível conhecer o valor da resistência necessária para atuar juntamente com os LEDs e transistores usados para amplificar a corrente, seguindo o *datasheet* do componente.

3.2 Montagem da fechadura eletromecânica

A fechadura eletromecânica é construída a partir de diversos componentes incluindo o motor de passo, *driver* para o uso do mesmo, referidos nas subseções 2.3.4 e 2.3.5 respectivamente. O motor de passo escolhido para a construção da fechadura eletromecânica é o modelo 28BYJ-48 juntamente com o *driver* ULN2003. Alguns materiais de plastico são utilizados na construção da fechadura eletromecânica, sendo uma pequena peça dentada

em formato circular fixada no motor de passo. A figura 15 mostra a peça acoplada ao motor.

Figura 15 – Peça de plástico dentada acoplada ao motor de passo



Fonte: Autor

A peça secundária montada para trabalhar juntamente com o motor de passo realizando movimentos horizontais através da rotação é formada por pequenas partes de plástico. Há uma pequena barra de ferro fixada na peça de plástico que será acoplada ao motor de passo e outra colada na barra com um furo na extremidade. A figura 16 mostra a peça secundária montada.

Figura 16 – Peça secundária montada para trabalhar com o motor de passo



Fonte: Autor

O protótipo é montado em formato de caixa quadrada com peças de compensado, tendo uma pequena abertura na lateral para a movimentação da peça secundária acoplada ao motor de passo. O tamanho do protótipo é relevante aos componentes fixados. O compensado utilizado tem a espessura de 1 cm. Com o auxílio de uma serra elétrica iniciou os cortes do compensado. A figura 17 mostra as peças retiradas do compensado.

Figura 17 – Peças de compensado para a montagem do protótipo



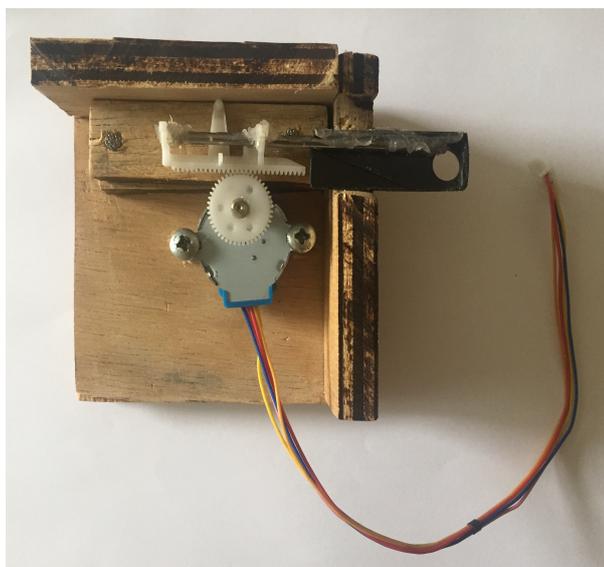
Fonte: Autor

As peças retiradas do compensado são fixadas por pregos e parafusos. As etapas do procedimento da montagem do protótipo são sequenciadas abaixo:

1. Fixar a parte superior na extremidade da base com um prego sem deixar elevações, apenas a margem de 1 cm em uma das laterais
2. Fixar a parte confeccionada com um prego na lateral com 1 cm restante para movimentação da peça secundária
3. Fixar as duas partes menores e iguais uma sobre a outra com dois pregos, na diagonal criada a partir das etapas anteriores
4. Posicionar a peça secundária sobre as partes menores e iguais encostando na extremidade superior para que não desloque na movimentação
5. Posicionar o motor de passo na base com o componente dentado fixado em direção a extremidade onde se encontra a parte confeccionada de modo que entre em contato com a peça secundária gerando o movimento horizontal, após a rotação
6. Fixar o motor de passo com dois parafusos, após o procedimento anterior
7. Fixar a lateral e a parte inferior pendente com dois pregos
8. Fixar a tampa do protótipo com dois parafusos, para a necessidade de futuras manutenções ou modificações internas

O protótipo tem 6 cm de altura, 10 cm de largura e comprimento. A figura 18 mostra o protótipo pronto e aberto frontalmente.

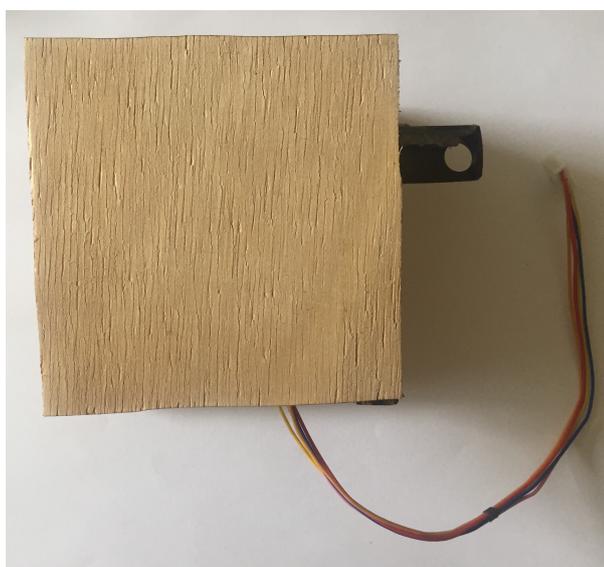
Figura 18 – Protótipo pronto e aberto frontalmente



Fonte: Autor

A figura 19 mostra o protótipo pronto e fechado.

Figura 19 – Protótipo pronto e fechado frontalmente



Fonte: Autor

A utilização deste protótipo é para realizar o controle do ferrolho, ou seja, o dispositivo criado trabalha em conjunto com o equipamento mecânico.

O ferrolho é fixado próximo ou sobre a porta, dependendo da situação. Ao enviar o comando necessário para ativação do protótipo, este agirá sobre o ferrolho alternando entre aberto e fechado. A figura 20 mostra o ferrolho utilizado com o protótipo.

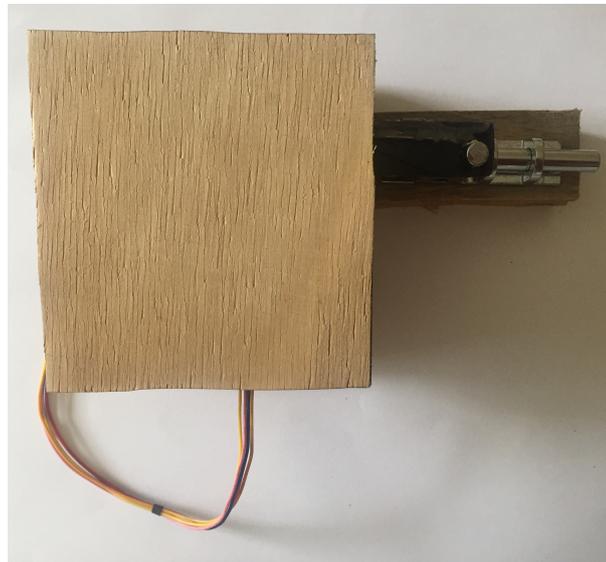
Figura 20 – Ferrolho utilizado com o protótipo



Fonte: Autor

O desenvolvimento final do protótipo juntamente com o ferrolho é mostrado na figura 21.

Figura 21 – Protótipo pronto testado com o ferrolho



Fonte: Autor

3.3 Modulo de reconhecimento de voz

O modulo de reconhecimento de voz é um *shield*, utilizado no sistema projetado para acionar os dispositivos escolhidos pelo usuário. Primeiramente os comandos são gravados no modulo, juntamente com o arduino, a programação necessária e a biblioteca do *shield* referida na subseção 2.4.2.

Quando os comandos gravados pelo usuário são repetidos no raio de alcance do modulo e identificados, ações são tomadas de acordo com a programação desenvolvida.

O modulo de reconhecimento de voz escolhido para o projeto é o *shield* EasyVR versão 3.0, algumas características são descritas abaixo.

- 28 comandos independentes (SI) prontos para comandos básicos (em inglês, italiano, japonês, alemão, espanhol e francês).

- Suporta até 32 comandos principais (SD) em qualquer linguagem, bem como senhas via voz.
- Bibliotecas para arduino disponíveis.
- Conector para entrada de microfone.
- Saída de áudio que suporta alto falantes de 8ohm

Os possíveis 32 comandos a serem gravados são divididos entre os dispositivos. A quantidade de comandos que cada dispositivo controlado por um LED emissor infravermelho pode oferecer é mostrado mais a frente na tabela 3. A divisão dos 32 possíveis comandos com todos os dispositivos controlados pelo sistema é mostrado na tabela 2.

Tabela 2 – Dispositivos, quantidade de comandos utilizados dos 32 possíveis e comandos dados

Dispositivos	Quantidade de comandos gravados	Comandos
Televisão	8	LigarTV, DesligarTV, Aumentarvolume, Diminuirvolume, Canalmais, Canalmenos, Canal 1, Canal 2
Aparelho de som	7	Ligar som, Desligar som, Aumentar volume, Diminuir volume, Estação mais, Estação menos, Estação gravada
Ar-condicionado	2	Ligarar, Desligarar
Lampada	2	Ligar lâmpada, Desligar lâmpada
Fechadura eletromecânica	2	Abrir, Trancar

Fonte: Autor

O total de comandos utilizados pelos dispositivos são 21 e os outros 11 são usados na programação, escolhendo o dispositivo e voltando para o menu principal, caso necessário

Os comandos mostrados na tabela 2 é apresentado de forma individual nas seções posteriores desta monografia.

A figura 22 mostra o EasyVR 3.0.

Figura 22 – EasyVR 3.0



Fonte: Autor

3.4 Identificação de dispositivos controlados por infravermelho

Há diversos dispositivos que são controlados por controle remoto através do LED emissor infravermelho embutido, enviando vários comandos.

Os dispositivos que recebem comandos por emissores infravermelho são escolhidos na residência para o controle por portadores de deficiência física e/ou visual, tais como: aparelho de som, televisão, ar-condicionado, DVD, *home theater*, ventiladores controlados por controle remoto, entre outros seguindo a preferência do usuário.

A escolha dos dispositivos a serem controlados pelo LED emissor infravermelho que será feita pelo usuário são de acordo com a preferência, apenas deve seguir os requisitos na quantidade máxima de comandos a serem gravados, estar no alcance do sistema e utilizar o controle remoto ou qualquer outro equipamento que tenha um LED emissor infravermelho enviando os comandos.

Os dispositivos que podem ser controlados pelo LED emissor infravermelho escolhidos são: televisão, ar-condicionado e aparelho de som.

Os equipamentos de controle dos dispositivos (controles remotos) que trabalham enviando comandos pelos LEDs emissores infravermelho embutidos são mostrados na figura 23.

Figura 23 – Controles remotos



Fonte:Autor

A escolha destes dispositivos ocorre por serem os mais comuns nas residências.

O controle remoto tem a aparência modificada de acordo com o dispositivo levando em consideração diversos aspectos, porém a real importância é o LED emissor infravermelho conectado a placa estar funcionando para futuramente realizar a leitura e emitir os códigos dos comandos.

A quantidade de comandos disponíveis (considerando cada botão do controle remoto um comando) em cada dispositivo utilizado é mostrado na tabela 3.

Tabela 3 – Controles remotos e quantidade de comandos

Controle remoto	Quantidade de comandos
Televisão	38
Aparelho de som	17
Ar-condicionado	14

Fonte: Autor

Apenas uma fração destes comandos poderão ser gravados e utilizados no sistema, seguindo a preferência do usuário na escolha.

Estes dispositivos são utilizados para a realização do projeto em questão. É possível implementar a metodologia seguida anteriormente em diferentes dispositivos.

3.5 Sistema de controle de dispositivos

Como comentado nas seções e capítulos anteriores, são 3 tipos de conexões para o controle de dispositivos, sendo estas por : relé, motor de passo e LED emissor infravermelho.

O controle da fechadura eletromecânica, mencionada na subseção 2.5.1 e seção 3.2, é realizado através do motor de passo.

O controle da lâmpada é realizado pelos pulsos enviados ao relé. O circuito esquematizado do controle da lâmpada é encontrado na seção 3.5.6.

O controle dos dispositivos que utilizam infravermelho é realizado pelo LED emissor. A seleção de dispositivos compatíveis com este sistema é mostrada na seção 3.4.

Os comandos para controlar todos os dispositivos são dados por voz. O algoritmo geral é a união de todos montados individualmente.

O sistema de controle de dispositivos é a união dos circuitos montados separadamente, ligados ao arduino e o *shield* de reconhecimento de voz, juntamente com a programação completa concluída e carregada.

3.5.1 Decodificação de comandos infravermelho

Os dispositivos controlados pelos controles remotos com o LED emissor infravermelho embutido, deve ter os comandos lidos e os códigos anotados para utilizar na programação do sistema.

Os comandos mostrados na tabela 2 são transmitidos pelos controles remotos dos dispositivos e retransmitidos pelo LED emissor infravermelho do sistema. Cada comando é um botão do controle remoto, sendo este posicionado de modo que o emissor se comunique com o receptor. Os comandos tem códigos específicos realizar tais ações.

A leitura dos comandos enviados pelos controles remotos depende da marca do dispositivo, pois, isto altera o algoritmo utilizado com o emissor infravermelho. O algoritmo da leitura dos comandos exige a especificação da marca do dispositivo para emitir os códigos. Os códigos emitidos são especificamente para tal marca, sendo ineficaz para realizar a mesma ação em um dispositivo diferente ou mesmo de outra marca.

3.5.2 Circuito decodificador de comandos enviados por infravermelho

Antes da montagem do circuito decodificador de comandos é verificado o *datasheet* para ter conhecimento dos valores de tensão e corrente suportados pelo componente.

O componente utilizado é o TSOP 1738, de acordo com o *datasheet*, suporta a tensão entre -0.3V a 6V e a corrente máxima é 5mA, entre outros parâmetros

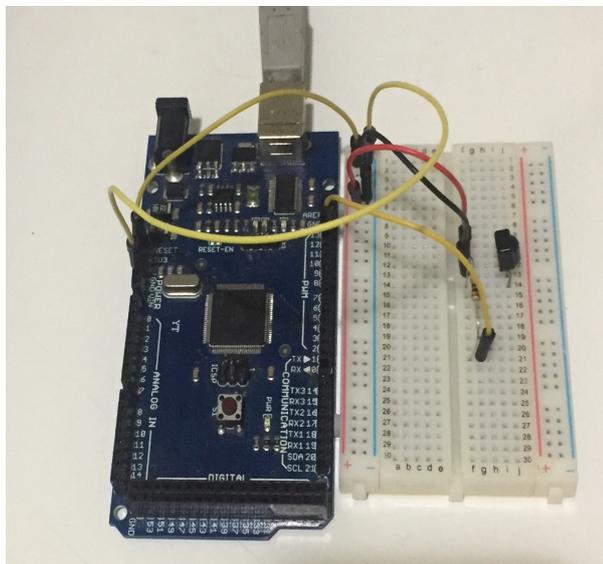
A montagem do circuito decodificador exige o uso de uma resistência para diminuir a corrente enviada.

Não é necessário montar o circuito decodificador de comandos enviados por infravermelho no proteus, pois, o mesmo não será embutido no sistema de controle de dispositivos. O circuito decodificador de sinais é usado apenas para obter os códigos dos controles remotos dos dispositivos.

Tendo conhecimento dos pinos do componente receptor infravermelho, é ligado ao *port* 11 do arduino, realizando a leitura ao apertar o botão de comando do controle remoto. O algoritmo utilizado para a leitura dos códigos é o algoritmo decodificador de sinais

infravermelhos encontrado no apêndice A. A figura 24 mostra o circuito decodificador de comandos infravermelho montado.

Figura 24 – Circuito decodificador de comandos enviados por infravermelho



Fonte: Autor

3.5.3 Controle de dispositivos pelo LED emissor infravermelho

O LED emissor infravermelho é capaz de controlar diversos dispositivos com o auxílio do arduino e outros componentes interligados.

Os dispositivos que utilizam infravermelho identificados na seção 3.4, serão os mesmos controlados pelo LED emissor infravermelho. Na subseção 3.5.1, os dispositivos controlados por infravermelho tiveram os comandos selecionados e os códigos obtidos. Nesta subseção, os códigos obtidos são usados para acionarem os dispositivos, realizando os comandos pelas frases ou palavras gravadas no módulo de reconhecimento de voz.

Usando um LED emissor infravermelho é capaz de controlar vários dispositivos, deste modo, o sistema tem apenas um para controlar a TV, o ar-condicionado e o aparelho de som, escolhidos neste projeto.

Neste trabalho os comandos gravados para controlar os dispositivos pelo emissor infravermelho são mostrados na tabela 2. Os comandos ligam ou desligam os dispositivos, aumenta ou diminui o volume, modifica os canais, entre outros.

O controle dos dispositivos pelo LED emissor infravermelho é dividido em 4 etapas, mostradas abaixo:

1. Identificar os dispositivos a serem controlados.
2. Dentre os comandos possíveis de cada dispositivo, selecionar os mais necessários, de acordo com a preferência do usuário.

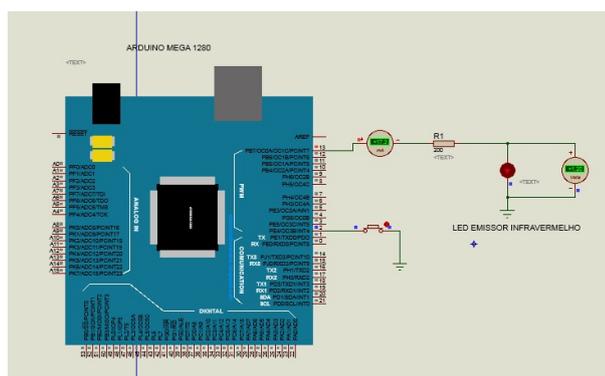
3. Obter os códigos dos comandos selecionados
4. Enviar os códigos obtidos pelo LED emissor infravermelho, acionando o comando no dispositivo.

As etapas foram divididas neste trabalho por subseções e explicadas detalhadamente. Esta subseção é o controle final dos dispositivos escolhidos, visando o envio dos códigos para realizar o comando em questão.

3.5.4 Circuito controlador de dispositivos pelo LED emissor infravermelho

O circuito controlador de dispositivos pelo LED emissor infravermelho é montado inicialmente no proteus, verificando os valores de tensão e corrente para que não exceda o máximo especificado pelo *datasheet* do componente, evitando futuras perdas. A figura 25 mostra o circuito virtual montado no proteus.

Figura 25 – Circuito controlador de dispositivos pelo LED emissor infravermelho no proteus



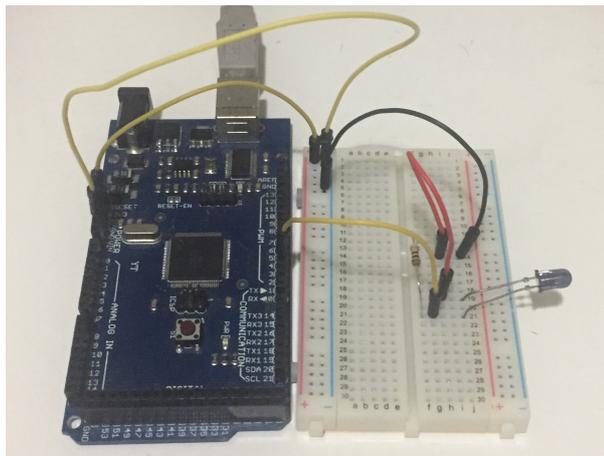
Fonte: Autor

O LED emissor infravermelho no proteus é substituído pelo convencional, pois, o funcionamento é o mesmo. Seguindo o *datasheet* do LED emissor infravermelho modelo TIL32 usado no projeto, sabe-se que a corrente máxima suportada é 20mA e a tensão 1.6V, deste modo, o circuito para atingir estes parâmetros, buscando a maior potência do componente para o melhor funcionamento. Usando o algoritmo teste 1 encontrado no apêndice B, é possível acionar o LED emissor infravermelho pela ativação do botão, todavia, utilizando a resistência correta, obtêm o valor desejado de corrente e tensão.

Nota-se que os valores de tensão e corrente são suportáveis tanto pelo arduino quanto pelo TIL32.

Os valores de tensão e corrente são buscados na montagem do circuito real. Os componentes usados no proteus são repetidos na montagem real. A figura 26 mostra o circuito real montado.

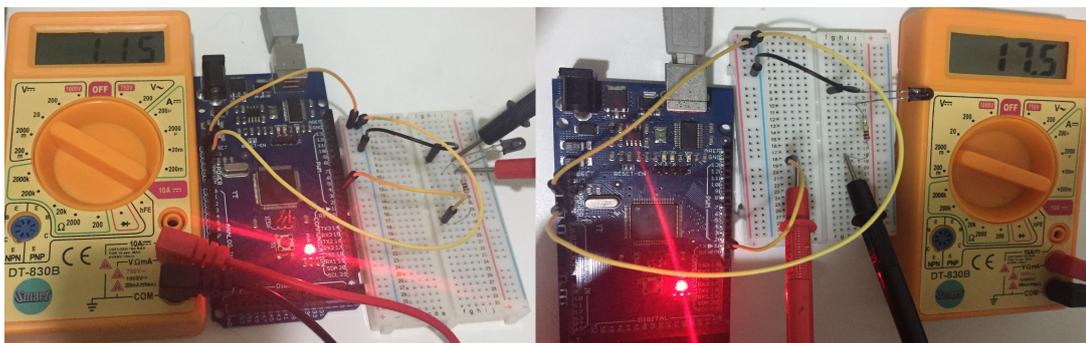
Figura 26 – Circuito controlador de dispositivos pelo LED emissor infravermelho



Fonte: Autor

É verificada a corrente e tensão no LED emissor infravermelho, esta é comparada com o valor obtido no proteus. A figura 27 mostra os valores de tensão e corrente obtidos.

Figura 27 – Tensão e corrente do circuito controlador de dispositivos pelo LED emissor infravermelho



Fonte: Autor

Os valores são aproximados, logo, o circuito real está dentro dos parâmetros.

3.5.5 Controle da lâmpada

A lâmpada é acionada por pulsos elétricos enviados pelo arduino para o relé, fazendo com que a ative o circuito secundário através da troca de contato elétrico, onde, tem o funcionamento detalhado na subseção 2.3.3. O acionamento da lâmpada, anteriormente realizada por um interruptor, é modificado e tem a conexão no relé, conectado ao arduino e a rede elétrica da residência (110V ou 220V).

Os comandos utilizados para o controle da lâmpada são: "ligar lâmpada" e "desligar lâmpada", O comando "ligar lâmpada" envia pulsos para o relé ativando o dispositivo, todavia, o comando "desligar lâmpada" atua inversamente. Os comandos são referidos na tabela 2.

Há dois algoritmos utilizados no controle da lâmpada, um para testes e outro para o circuito real. O algoritmo para teste é usado no proteus, realizando a mesma função do circuito real, porém, sem os comandos enviados por voz. O algoritmo do circuito real é usado juntamente com o sistema final do projeto, onde, se tem o controle dos dispositivos por voz.

Inicialmente o controle da lâmpada é realizado individualmente para teste, após finalizar esta etapa é possível fazer a unificação de todos os dispositivos, através dos circuitos montados.

3.5.6 Circuito acionador da lâmpada

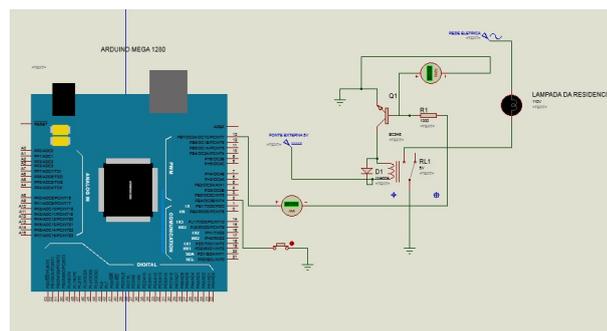
O circuito acionador da lâmpada é formado pelo modulo rele que engloba uma série de componentes necessários para não exceder o valor de corrente e tensão do arduino. O modulo rele é utilizado juntamente com arduino em diversos projetos, neste é para o acionamento da lâmpada por voz.

O circuito acionador da lâmpada é inicialmente montado no proteus para verificar os valores de tensão e corrente, evitando perdas de componentes ou mal funcionamento do sistema final.

Para exemplificação do projeto o botão será o acionador da lâmpada, todavia, no projeto real os dispositivos serão acionados por voz. O algoritmo utilizado é o algoritmo de teste 1 encontrado no apêndice B.

A figura 28 mostra o circuito virtual montado no proteus e o acionamento da lâmpada realizado pelo botão, neste caso não está ativado.

Figura 28 – Circuito acionador da lâmpada no proteus desativado

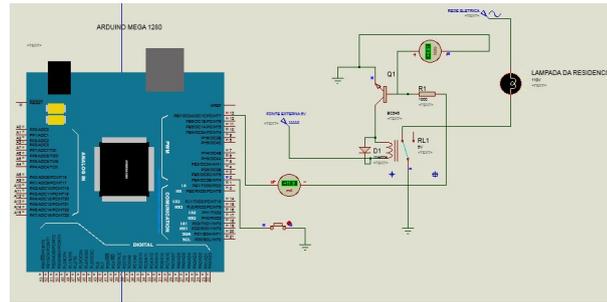


Fonte: Autor

Nesta primeira etapa o circuito está desligado, ou seja, o botão não é acionado.

A figura 29 mostra o circuito virtual montado no proteus e o acionamento da lâmpada realizado pelo botão, neste caso está ativado.

Figura 29 – Circuito acionador da lâmpada no proteus ativado



Fonte: Autor

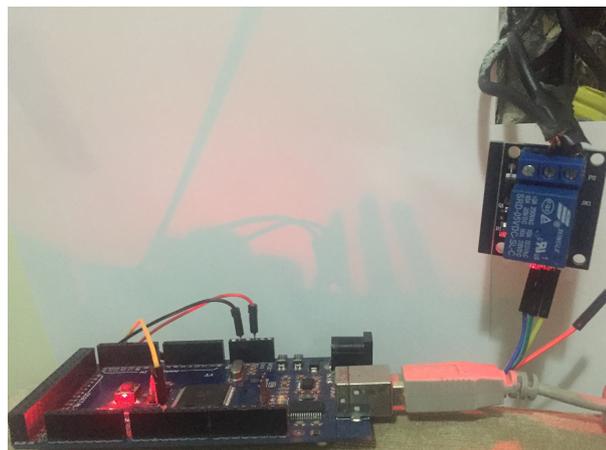
No proteus o modulo relé mencionado na subseção 2.3.3 é montado usando os componentes acoplados na placa. No circuito real, a utilização do modulo relé é mais prática e fácil

Nesta segunda etapa o circuito está ligado, ou seja, o botão é acionado. A corrente que flui do arduino para o transistor é mostrada e não é superior ao valor máximo permitido. O relé está funcionando de acordo com o especificado, a lâmpada é acionada ao ativar o botão.

É iniciada a montagem real do circuito seguindo os componentes com os valores mostrados no proteus. O modulo rele possui todos os componentes necessários, facilitando o uso e a montagem do circuito.

A figura 30 mostra o circuito acionador da lâmpada real montado.

Figura 30 – Circuito acionador da lampada

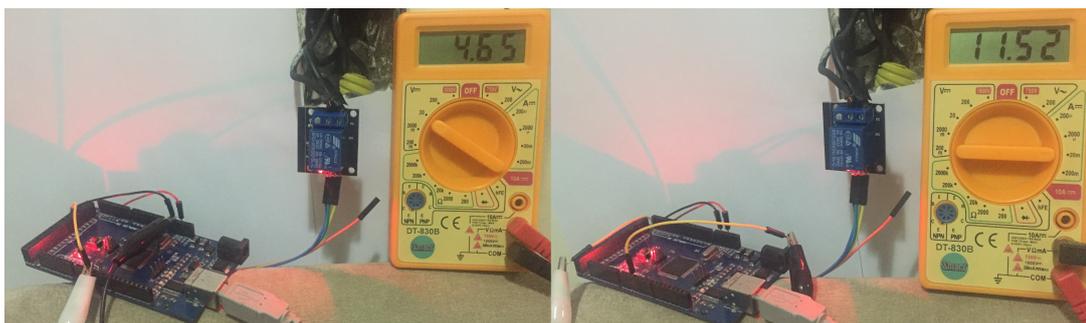


Fonte: Autor

O valor da corrente e tensão são testados para verificar a veracidade dos dados obtidos pelo proteus e por medida de segurança, pois há a possibilidade de ter problemas na montagem real do circuito.

A figura 31 mostra os valores obtidos na montagem e funcionamento do circuito real.

Figura 31 – Tensão e corrente do circuito acionador da lampada



Fonte: Autor

Os valores são aproximados e o circuito é aprovado.

3.5.7 Controle da fechadura eletromecânica

A fechadura eletromecânica é controlada pelos pulsos elétricos enviados para o motor de passo, através da programação criada e compilada no arduino. Os componentes necessários são acoplados no protótipo montado e referido na seção 3.2. O protótipo trabalha juntamente com o ferrolho fixado na porta ou nas proximidades.

O protótipo é fixado próximo ao ferrolho. A parte externa do protótipo, onde há um buraco na lateral, é fixado na base do ferrolho.

Os comandos utilizados para a fechadura são: Abrir e trancar, mencionados na tabela 2. Quando o comando "abrir porta" é mencionado dentro do alcance do sistema de controle e é detectado, o motor de passo gira no sentido anti-horário e a peça dentada acoplada puxa a base do ferrolho para esquerda, deste modo, não há interferências para a abertura da porta, todavia, o comando "trancar porta" realiza o movimento inverso, girando no sentido horário, impossibilitando a movimentação da porta.

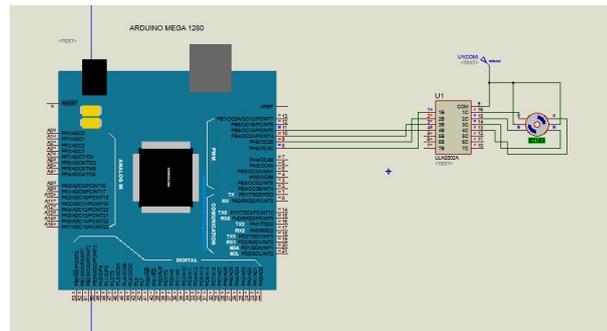
O controle da fechadura eletromecânica é realizado individualmente, ou seja, não há ligações com outros dispositivos controlados. Esta união é realizada após os teste e a verificação da funcionalidade do controle da fechadura.

3.5.8 Circuito controlador da fechadura eletromecânica

O circuito controlador da fechadura eletromecânica é formado pelo *driver* ULN2003 que engloba todos os componentes necessários para comunicação com o arduino fazendo com que não ultrapasse os limites de tensão e corrente. O *driver* ULN2003 é explicado detalhadamente na subseção 2.3.5.

A montagem do circuito é iniciada pelo proteus por segurança. A figura 32 mostra o circuito virtual controlador da fechadura eletromecânica montado no proteus.

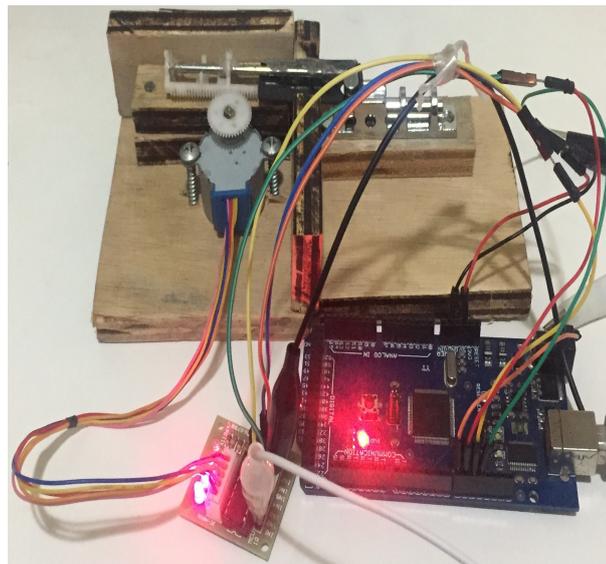
Figura 32 – Circuito controlador da fechadura eletromecânica no proteus



Fonte: Autor

O circuito montado no proteus é essencial para montagem real. No proteus mostra as ligações do arduino, *driver* ULN2003 e o motor de passo 28BYJ-48. No circuito real, o *driver* ULN2003 é de maior praticidade na utilização, pois, as conexões estão prontas e têm LEDs demonstrativos ao ativar o motor de passo 28BYJ-48. A figura 33 mostra o circuito real controlador da fechadura eletromecânica montado.

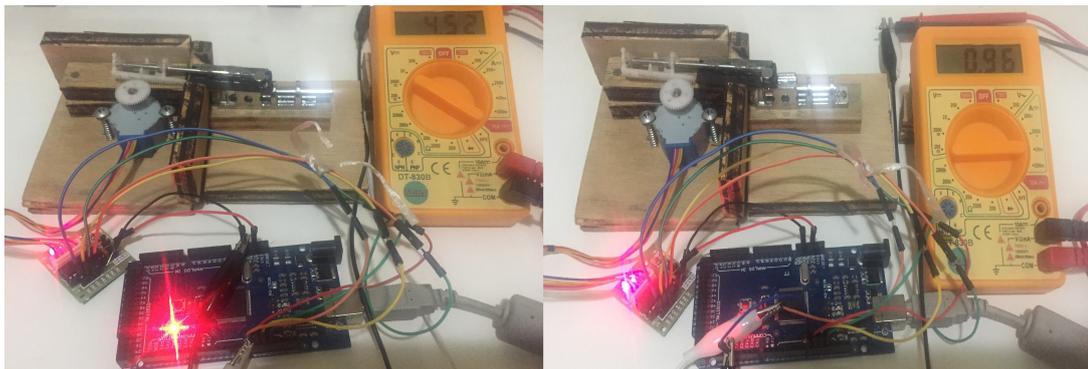
Figura 33 – Circuito controlador da fechadura eletromecânica



Fonte: Autor

Os procedimentos realizados no proteus são repetidos no circuito real e os valores de tensão e corrente são calculados, afim de verificar se não há erros na ligação e deste modo, não forçar o circuito. A figura 34 mostra o circuito controlador da fechadura eletromecânica montado e os valores de tensão e corrente obtidos.

Figura 34 – Tensão e corrente do circuito controlador da fechadura eletromecânica



Fonte: Autor

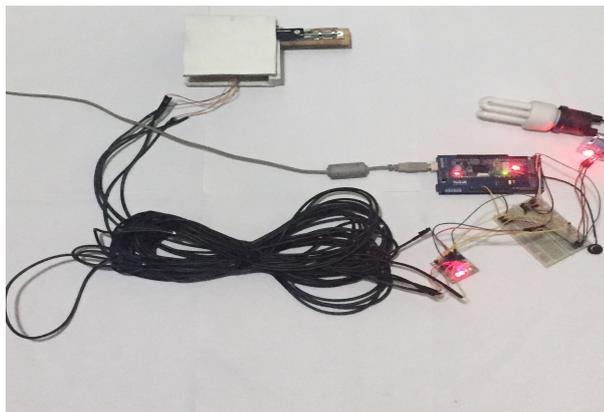
O algoritmo de controle da fechadura eletromecânica é compilado para teste, sendo este o algoritmo teste 2 encontrado no apêndice C, porém, apenas para o fechamento deste circuito. O algoritmo final é desenvolvido através da união dos anteriores.

3.6 Unificação dos circuitos de controle

Os circuitos montados nas subseções 3.5.4, 3.5.6 e 3.5.8 são unidos para formar o sistema de controle de dispositivos, controlando todos por voz. Os dispositivos controlados por infravermelho, pelo relé e *driver* ULN2003 são conectados no arduino juntamente com o *shield* EasyVR ou módulo de reconhecimento de voz. O LED emissor infravermelho é conectado na porta 9, enquanto o relé na porta 44 e o *driver* ULN2003 estão nas portas 50 até 53.

Os dispositivos conectados fisicamente ao sistema são ligados por fios finos de espessura 15mm. A fechadura eletromecânica e o relé para controle da lâmpada são estes. A figura 35 mostra a conexão da lâmpada e a fechadura no arduino com módulo de reconhecimento de voz.

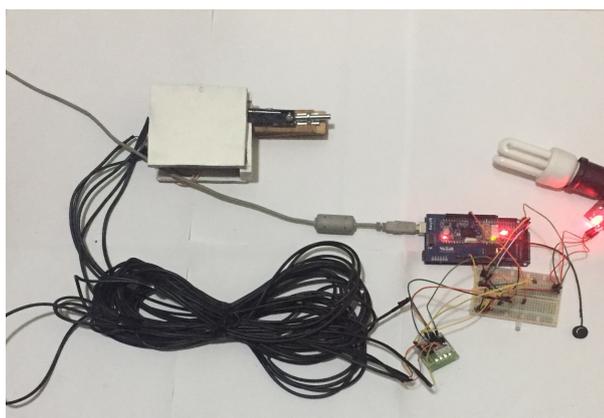
Figura 35 – Lâmpada e fechadura conectados no arduino com o módulo de reconhecimento de voz



Fonte: Autor

O LED emissor infravermelho não necessita de interação física para controlar os dispositivos, apenas deve posicioná-lo dentro do alcance de comunicação. A posição final do sistema tem a interação com os 3 dispositivos a serem controlados pelo LED emissor infravermelho. A figura 36 a conexão final do sistema, com o LED emissor infravermelho, a fechadura eletromecânica e o relé para controle da lâmpada no arduino juntamente com o módulo de reconhecimento de voz.

Figura 36 – Lâmpada, fechadura e LED emissor infravermelho conectados no arduino com o módulo de reconhecimento de voz



Fonte: Autor

O algoritmo final encontrado no apêndice D do sistema de controle de dispositivos é desenvolvido a partir da união dos algoritmos individuais dos dispositivos, acionando todos pelos comandos gravados no módulo de reconhecimento de voz ou *shield* EasyVR versão 3.0. O usuário habilita a ação do dispositivo repetindo qualquer comando gravado e apresentado na tabela 2.

3.7 Protótipo para armazenamento da central de controle e dos componentes

A construção do protótipo de armazenamento da central de controle e componentes é feita por questão estética e organização. A central é fixada na caixa de formato retangular feita em compensado de espessura 10mm. Um pequeno espaço é aberto na parte superior, o qual, o LED emissor infravermelho é colocado para comunicação com os dispositivos sem interferências. Outros pequenos espaços são feitos para acoplar LEDs de cor verde utilizando as portas 45 até 49 e serve para indicar se o dispositivo está ligado e o maior para posicionar o comunicador do módulo de reconhecimento de voz. Na lateral há espaços para passagem dos fios conectados aos dispositivos fisicamente e outro pequeno buraco para passagem do fio USB de comunicação com o arduino e o módulo de reconhecimento de voz. Também há o espaço na lateral para conectar o arduino a bateria, caso necessário.

Primeiramente a peça de compensado é cortada em partes menores para construção do protótipo. O protótipo tem 30mm de altura, 20mm de comprimento e 10mm de largura. A figura 37 mostra as peças retiradas do compensado.

Figura 37 – Peças retiradas do compensado para montagem do protótipo



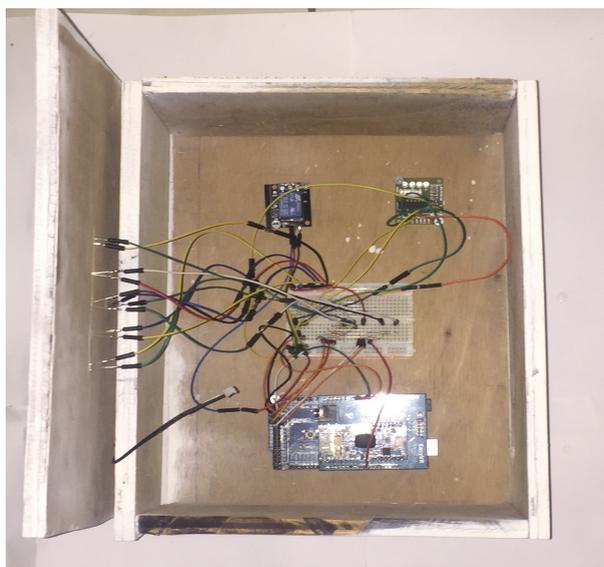
Fonte: Autor

A montagem do protótipo consiste nas seguintes etapas:

1. Fixar a base com a parte superior usando 2 pregos em cada extremidade
2. Fixar a base com a lateral direita usando 2 pregos em cada extremidade
3. Fixar a base com a lateral esquerda usando 2 pregos em cada extremidade
4. Fixar a base com a parte inferior usando 2 pregos em cada extremidade
5. Fixar os componentes (isto inclui o LED emissor infravermelho que tem um local específico), a central de controle (arduino e modulo de reconhecimento de voz(EasyVR)), organizadamente.
6. Passar os fios pelo espaço aberto para ligar os dispositivos no arduino com o modulo de reconhecimento de voz (EasyVR).
7. Fixar a parte frontal com 2 parafusos, caso seja necessário abrir e realizar futuros reparos ou modificações.

A figura 38 mostra o protótipo pronto sem a parte frontal, mostrando a organização dos componentes e a central de controle.

Figura 38 – Protótipo pronto aberto



Fonte: Autor

A figura 39 mostra o protótipo pronto com a parte frontal, mostrando a parte final do projeto.

Figura 39 – Protótipo pronto fechado



Fonte: Autor

Deste modo, o sistema de controle de dispositivos está pronto.

3.8 Teste prático

A unificação dos circuitos, a montagem dos protótipos, a compilação do algoritmo, finalizou o sistema de controle de dispositivos, deste modo, a ultima etapa do projeto é testar o sistema e verificar o funcionamento geral.

O primeiro teste consiste em acionar a TV utilizando os comandos gravados e apresentados na tabela 2. Deste modo, o teste é valido para verificar se o LED emissor infravermelho ao enviar os códigos pelo comando emitido aciona a TV.

Depois da aprovação do primeiro teste, é feito o segundo teste nos comandos do aparelho de som, da mesma forma que feita anteriormente. Em seguida é testado os comandos do ar-condicionado repetindo as mesmas etapas anteriores.

Após a aprovação dos testes anteriores, os dispositivos ligados diretamente ao sistema de controle são testados. Os comandos gravados para controlar a lâmpada são repetidos pelo usuário testando o funcionamento deste circuito.

Por ultimo, o teste é realizado na fechadura eletromecânica, verificando a atuação do ferrolho repetindo os comandos gravados para o controle do mesmo.

As etapas anteriores sendo aprovadas, há a ultima verificação na corrente e tensão do circuito enquanto os dispositivos estão atuando. Esta verificação é por segurança, caso tenha ocorrido erros na montagem do sistema de controle de dispositivos.

4 Considerações finais

A tecnologia assistiva visa ajudar os portadores de diversas deficiências. A tecnologia é usada em prol dos portadores de deficiência, aumentando a habilidade funcional, restringida.

O projeto é dividido em 8 etapas consideradas objetivos específicos, onde, estas foram seguidas rigorosamente para o término do projeto. O primeiro objetivo específico é realizar o enredo bibliográfico do tema, buscando outros projetos parecidos e o estudo aprofundado sobre os componentes utilizados no projeto, entendendo o funcionamento e o uso. Em seguida os outros objetivos específicos são seções e subseções do projeto, sendo estes: Obter a leitura dos comandos enviados pelos controles remotos infravermelho, criar o protótipo de controle da fechadura eletromecânica, montar os circuitos para o controle da fechadura eletromecânica, dos dispositivos infravermelho e acionador da lâmpada, desenvolver o algoritmo para o funcionamento em conjunto dos circuitos controladores dos dispositivos, criar o protótipo para armazenamento da central de controle e dos componentes, unir os circuitos juntamente com os componentes necessários e finalmente testar o sistema. Seguindo rigorosamente cada objetivo citado, o desenvolvimento do projeto final é concretizado.

O desenvolvimento do algoritmo não teve dificuldades, utilizando os conhecimentos obtidos nas matérias do curso de engenharia de controle e automação e os estudos feitos no início do projeto, facilitaram a programação. O algoritmo necessitou o uso de algumas bibliotecas para o funcionamento eficaz e preciso do projeto, como previsto.

Os comandos foram gravados através do programa chamado *EasyVR Commander* versão 3.11.0, realizando as alterações necessárias no módulo de reconhecimento de voz. Após a gravação dos comandos, o programa cria um algoritmo com tudo gravado dentro do EasyVR. As alterações são feitas de acordo com o projeto.

Após os inúmeros testes envolvendo os cálculos de tensão e corrente mostrados nas figuras 27, 31 e 34, notou-se que o projeto está nos parâmetros desejados, dando continuidade às etapas posteriores sem afetar o funcionamento do sistema ou diminuir a vida útil dos componentes.

O teste final do projeto verificou a utilização do sistema no geral, sem o uso dos controles remotos, da chave ou interruptor dos dispositivos. A facilidade gerada para os portadores de deficiência física e/ou visual é satisfatória para o projeto.

Os portadores de deficiência física e/ou visual têm auxílio do sistema de controle de dispositivo para realizar atividades diárias. O projeto final visa a facilidade na comunicação entre os dispositivos e os usuários.

Inicialmente o projeto almejava apenas 3 dispositivos utilizando o *shield* de reconhecimento de voz da versão anterior, ou seja, EasyVR 2.0 pelo baixo custo, mas o projeto

exigiu uma quantidade maior de comandos e a versão atual disponibiliza isto. A quantidade de dispositivos fora alterada para 5, necessitando agora maior capacidade do ATmega328P principalmente para o controle por infravermelho, deste modo, fora trocado para o ATmega1280.

O sistema de controle de dispositivos é satisfatório e totalmente viável, mostrando uma forma alternativa com menor custo comparado a compra de apenas um dispositivo apresentando esta função. A independência proporcionada para os portadores de deficiência física e/ou visual é relevante comparado ao custo do projeto.

4.1 Custo final do projeto

O projeto tem um custo equivalente aos componentes, compensados e cabos utilizados. Para diminuir o custo, fora utilizado o compensado guardado na residência para montagem do protótipo da fechadura eletromecânica e recipiente para organizar os componentes. O projeto é escrito e desenvolvido de maneira prática e fácil para qualquer pessoa que almeje repetir o procedimento realizado, deste modo, o cálculo dos custos são feitos buscando o valor aproximado gasto incluindo os objetos que não foram comprados, analisando o valor de mercado.

A tabela 4 mostra os objetos utilizados e o valor aproximado gasto ou médio encontrado no mercado.

Tabela 4 – Custo total dos objetos utilizados no sistema

Objeto	Valor
Componentes eletrônicos	R\$ 12,00
Cabos e jumpers	R\$ 25,00
Arduino Mega	R\$ 90,00
Shield EasyVR 3.0	R\$ 250,00
Chapa de compensado	R\$ 80,00
Total	R\$ 457,00

Fonte: Autor

Os objetos foram comprados no site Mercado Livre, pelo baixo custo e boa qualidade oferecida, ou seja, os valores têm o frete incluso, mostrando o total gasto em cada um.

4.2 Trabalhos futuros

Para realização de trabalhos futuros ligados ao tema são feitas indicações iniciais para implementar e melhorar o projeto. São 4 indicações do autor deste projeto para trabalhos futuros, mostradas abaixo

- Montagem da placa de controle, embutindo todos os componentes do projeto

- Diminuição do protótipo de armazenamento de componentes e da plataforma de controle
- Movimentação livre do LED emissor infravermelho para caso haja modificação do local dos dispositivos controlados por este
- Resposta por voz do sistema para o usuário na utilização dos comandos.

4.3 Trabalhos relacionados

Há monografias que foram comparadas tanto na parte escrita quanto prática, relacionando os temas e verificando a metodologia utilizada para ter a base deste projeto.

A parte escrita e prática teve auxílio do TCC do Álvaro José Belota de Oliveira Saraiva, formado em 2015 pela Universidade do Estado do Amazonas, cujo tema é sistema eletrônico para locomoção de pessoas com deficiência visual em ambientes estruturados. Na maioria fora seguida a parte escrita por estar bem estruturado de acordo com o orientador do autor deste projeto. O tema é semelhante por utilizar arduino e auxiliar pessoas com deficiência visual.

No projeto do Álvaro Belota é possível auxiliar deficientes visuais a encontrar obstáculos e esquivar caso necessário, o presente trabalho auxilia de forma contínua incluindo os deficientes físicos no controle dispositivos. Ambos auxiliam pessoas com deficiência.

O segundo projeto utilizado fora do Ciro Nascimento Lopes, formado em 2015 pela Universidade do Estado do Amazonas, sendo na parte prática pela semelhança do tema, cujo é Automação Residencial com a Utilização de Microcontrolador Associado a Dispositivos Móveis. Neste projeto é possível controlar diversos dispositivos por *bluetooth*, utilizando celulares, *tablets*, etc. O presente projeto é um aprimoramento, sendo usado para auxílio de deficientes e fornecendo os comandos por voz eficazmente.

APÊNDICE A – Algoritmo decodificador de sinais infravermelhos

```
#include <IRremote.h>

int RECV_PIN = 11;

IRrecv irrecv(RECV_PIN);

decode_results results;

void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn();
}

void loop() {
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume();
  }
  delay(100);
}
```

APÊNDICE B – Algoritmo teste 1

```
const int buttonPin = 2;
const int ledPin = 13;

int buttonState = 0;

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}

void loop() {
  buttonState = digitalRead(buttonPin);

  if (buttonState == HIGH) {
    digitalWrite(ledPin, HIGH);
  }
  else {
    digitalWrite(ledPin, LOW);
  }
}
```

APÊNDICE C – Algoritmo teste 2

```
#include <Stepper.h>

Stepper fechadura(2048,8,9,10,11);

void setup() {
  fechadura.setSpeed(10);
}

void loop() {
  fechadura.step(750);
  delay(1000);
  fechadura.step(-750);
  delay(1000);
}
```

APÊNDICE D – Algoritmo final

```

#include <IRremote.h>
IRsend irsend;
#include <Stepper.h>
Stepper fechadura(2048,50,51,52,53);
#include "Arduino.h"
#if !defined(SERIAL_PORT_MONITOR)
  #error "Arduino version not supported. Please update your IDE to the latest version."
#endif

#if defined(SERIAL_PORT_USBVIRTUAL)
  #define port SERIAL_PORT_HARDWARE
  #define pcSerial SERIAL_PORT_USBVIRTUAL
#else
  #include "SoftwareSerial.h"
  SoftwareSerial port(12, 13);
  #define pcSerial SERIAL_PORT_MONITOR
#endif

#include "EasyVR.h"

EasyVR easyvr(port);

enum Groups
{
  GROUP_0 = 0,
  GROUP_1 = 1,
  GROUP_2 = 2,
  GROUP_3 = 3,
  GROUP_4 = 4,
  GROUP_5 = 5,
  GROUP_6 = 6,
};

enum Group0
{
  G0_ARDUINO = 0,
};

enum Group1
{
  G1_LIGARTV = 0,
  G1_DESLIGARTV = 1,
  G1_AUMENTARVOLUME = 2,
  G1_DIMINUIRVOLUME = 3,
  G1_CANALMAIS = 4,
  G1_CANALMENOS = 5,
};

```

```
G1_CANAL_1 = 6,  
G1_CANAL_2 = 7,  
G1_VOLTAR = 8,  
};  
  
enum Group2  
{  
    G2_LIGARSOM = 0,  
    G2_DESLIGARSOM = 1,  
    G2_AUMENTARSOM = 2,  
    G2_DIMINUIRSOM = 3,  
    G2_ESTACAOMAI = 4,  
    G2_ESTACAOMENOS = 5,  
    G2_ESTACAOGRAVADA = 6,  
    G2_VOLTAR = 7,  
};  
  
enum Group3  
{  
    G3_LIGARAR = 0,  
    G3_DESLIGARAR = 1,  
    G3_VOLTAR = 2,  
};  
  
enum Group4  
{  
    G4_LIGARLUZ = 0,  
    G4_DESLIGARLUZ = 1,  
    G4_VOLTAR = 2,  
};  
  
enum Group5  
{  
    G5_ABRIRPORTA = 0,  
    G5_TRANCARPORTA = 1,  
    G5_VOLTAR = 2,  
};  
  
enum Group6  
{  
    G6_TV = 0,  
    G6_SOM = 1,  
    G6_ARCONDICIONADO = 2,  
    G6_LAMPADA = 3,  
    G6_FECHADURA = 4,  
};  
  
int8_t group, idx;  
int T = 0;
```

unsigned int LIGARSOM2[67] = {4600,4350, 650,450, 700,400, 700,400, 700,450, 650,1550, 650,450, 650,450, 650,500, 650,450, 700,400, 650,450, 650,450, 700,1550, 650,450, 650,450, 650,450, 700,400, 700,1550, 650,1550, 700,1550, 600,1600, 650,450, 700,400, 650,500, 650,1550, 700,400, 650,450, 700,450, 650,450, 650,1550, 650,1600, 650,1550, 650};

unsigned int DESLIGARSOM2[67] = {4600,4350, 650,450, 700,400, 700,400, 700,450, 650,1550, 650,450, 650,450, 650,500, 650,450, 700,400, 650,450, 650,450, 700,1550, 650,450, 650,450, 650,450, 700,400, 700,1550, 650,1550, 700,1550, 600,1600, 650,450, 700,400, 650,500, 650,1550, 700,400, 650,450, 700,450, 650,450, 650,1550, 650,1600, 650,1550, 650};

unsigned int AUMENTARSOM2[67] = {4600,4300, 650,450, 700,450, 650,450, 650,450, 650,1550, 700,450, 650,450, 650,450, 650,450, 700,400, 700,450, 650,450, 650,1550, 650,450, 700,450, 650,450, 650,1550, 650,1550, 700,1550, 650,450, 650,1550, 650,500, 650,450, 650,450, 650,450, 700,400, 650,500, 650,1550, 650,450, 700,1550, 650,1550, 650,1550, 650};

unsigned int DIMINUIRSOM2[67] = {4600,4350, 650,450, 650,450, 650,450, 700,400, 700,1550, 650,450, 650,450, 700,400, 700,450, 650,450, 650,450, 650,450, 650,1600, 650,450, 650,450, 650,450, 650,450, 700,1550, 650,1550, 650,450, 700,1550, 600,500, 650,450, 650,450, 700,1550, 600,500, 650,450, 650,1550, 700,450, 600,1600, 650,1550, 700,1550, 650};

unsigned int ESTACAOMAS2[67] = {4600,4300, 700,450, 600,500, 650,450, 650,450, 650,1600, 650,450, 650,450, 650,450, 650,450, 700,400, 650,500, 600,500, 650,1550, 700,400, 700,450, 600,500, 650,1550, 650,1600, 600,1600, 650,450, 700,400, 700,450, 600,500, 650,450, 650,450, 700,450, 650,1550, 650,1550, 700,1550, 650,1550, 650,1550, 700};

unsigned int ESTACAOMENOS2[67] = {4600,4300, 650,500, 650,450, 650,450, 650,450, 650,1600, 600,500, 650,450, 650,450, 650,450, 650,500, 650,450, 650,450, 650,1550, 650,500, 600,500, 650,450, 650,450, 650,1550, 700,1550, 650,450, 650,450, 650,450, 650,500, 600,500, 650,1550, 650,450, 700,450, 600,1600, 650,1550, 700,1550, 650,1550, 650,1600, 650};

unsigned int ESTACAOGRAVADA2[67] = {4650,4300, 650,450, 650,450, 650,500, 600,500, 650,1550, 650,450, 650,450, 650,500, 600,500, 650,450, 650,450, 650,450, 650,1600, 650,1550, 650,450, 650,500, 600,1600, 650,450, 650,450, 650,500, 600,500, 650,450, 650,1550, 650,1600, 650,450, 650,1550, 650,1600, 600,1600, 650};

unsigned int LIGARAR2[199] = {4550,4300, 600,1550, 650,450, 650,1500, 600,1550, 650,450, 600,450, 650,1550, 650,400, 650,450, 650,1500, 700,400, 600,500, 650,1500, 700,1450, 650,400, 650,1550, 600,1550, 750,350, 600,450, 700,1450, 750,1400, 700,1500, 750,1400, 600,1550, 600,500, 650,1500, 650,1500, 600,500, 650,400, 750,350, 600,500, 650,400, 650,1550, 700,1450, 600,1550, 650,400, 800,300, 650,1500, 700,400, 600,500, 600,450, 650,450, 650,450, 600,1550, 650,1500, 600,500, 600,1550, 600,1550, 650,5550, 4500,4300, 650,1500, 600,500, 600,1550, 650,1500, 600,450, 650,450, 650,1500, 650,450, 600,500, 650,1500, 600,500, 600,450, 650,1500, 650,1500, 650,450, 650,1500, 700,1500, 650,400, 650,450, 600,1550, 650,1500, 650,1550, 600,1550, 750,1400, 650,450, 600,1550, 650,1500, 600,500, 600,450, 700,400, 650,450, 600,450, 650,1500, 650,1550, 600,1550, 650,400, 700,400, 650,1500, 650,450, 600,500, 650,400, 650,450, 700,400, 700,1450, 600,1550, 600,500, 650,1500, 600,1550, 650};

unsigned int DESLIGARAR2[199] = {4550,4300, 750,1400, 650,400, 650,1550, 600,1550, 600,450, 650,450, 650,1500, 650,450, 600,500, 650,1500, 650,400, 650,450, 650,1500, 750,1400, 650,450, 650,1500, 750,350, 650,1500, 650,1500, 650,1500, 700,1500, 600,450,

```
650,1500, 700,1500, 600,1550, 600,450, 700,400, 650,450, 600,450, 650,1550, 600,450,
700,400, 650,1500, 650,1500, 650,1500, 650,450, 700,400, 650,400, 650,450, 650,450,
600,500, 600,450, 650,450, 650,1500, 650,1500, 650,1500, 650,1500, 650,1550, 600,5550,
4550,4300, 700,1450, 650,400, 650,1550, 600,1550, 600,450, 700,400, 600,1550, 650,450,
650,450, 600,1550, 650,400, 650,450, 650,1500, 650,1500, 650,450, 650,1500, 700,400,
650,1500, 650,1500, 650,1500, 650,1550, 650,400, 650,1500, 650,1550, 650,1500, 600,450,
650,450, 650,450, 650,400, 650,1550, 650,400, 650,450, 650,1500, 700,1450, 700,1450,
650,450, 700,400, 650,400, 650,450, 650,450, 650,450, 600,450, 650,450, 600,1550,
650,1500, 650,1500, 650,1550, 600,1550, 650};
```

```
void setup()
{
  fechadura.setSpeed(10);
  pinMode(9, OUTPUT);
  pinMode(44, OUTPUT);
  pinMode(45, OUTPUT);
  pinMode(46, OUTPUT);
  pinMode(47, OUTPUT);
  pinMode(48, OUTPUT);
  pinMode(49, OUTPUT);
  digitalWrite(9,LOW);
  pcSerial.begin(9600);

  int mode = easyvr.bridgeRequested(pcSerial);
  switch (mode)
  {
    case EasyVR::BRIDGE_NONE:
      port.begin(9600);
      pcSerial.println(F("---"));
      pcSerial.println(F("Bridge not started!"));
      break;

    case EasyVR::BRIDGE_NORMAL:
      port.begin(9600);
      easyvr.bridgeLoop(pcSerial);
      pcSerial.println(F("---"));
      pcSerial.println(F("Bridge connection aborted!"));
      break;

    case EasyVR::BRIDGE_BOOT:
      port.begin(115200);
      easyvr.bridgeLoop(pcSerial);
      pcSerial.println(F("---"));
      pcSerial.println(F("Bridge connection aborted!"));
      break;
  }

  while (!easyvr.detect())
  {
    Serial.println("EasyVR not detected!");
  }
}
```

```

    delay(1000);
}

easyvr.setPinOutput(EasyVR::IO1, LOW);
Serial.println("EasyVR detected!");
easyvr.setTimeout(5);
easyvr.setLanguage(0);

group = EasyVR::TRIGGER;
}

void action();

void loop()
{
if (easyvr.getID() < EasyVR::EASYVR3)
easyvr.setPinOutput(EasyVR::IO1, HIGH);

Serial.print("Say a command in Group ");
Serial.println(group);
easyvr.recognizeCommand(group);

do
{
}
while (!easyvr.hasFinished());

if (easyvr.getID() < EasyVR::EASYVR3)
easyvr.setPinOutput(EasyVR::IO1, LOW); // LED off

idx = easyvr.getWord();
if (idx >= 0)
{
return;
}
idx = easyvr.getCommand();
if (idx >= 0)
{
uint8_t train = 0;
char name[32];
Serial.print("Command: ");
Serial.print(idx);
if (easyvr.dumpCommand(group, idx, name, train))
{
Serial.print(" = ");
Serial.println(name);
}
}
else
Serial.println();
easyvr.playSound(0, EasyVR::VOL_FULL);

```

```

    action();
}
else
{
    if (easyvr.isTimeout())
        Serial.println("Timed out, try again...");
    int16_t err = easyvr.getError();
    if (err >= 0)
    {
        Serial.print("Error ");
        Serial.println(err, HEX);
    }
}
}
}

```

```

void action()
{
    switch (group)
    {
        case GROUP_0:
            switch (idx)
            {
                case G0_ARDUINO:
                    group = GROUP_6;
                    break;
            }
            break;
        case GROUP_1:
            switch (idx)
            {
                case G1_LIGARTV:
                    irsend.sendRC6(0xC, 20);
                    digitalWrite(49, HIGH);
                    break;
                case G1_DESLIGARTV:
                    irsend.sendRC6(0xC, 20);
                    digitalWrite(49, LOW);
                    break;
                case G1_AUMENTARVOLUME:
                    irsend.sendRC6(0x10, 20);
                    break;
                case G1_DIMINUIRVOLUME:
                    irsend.sendRC6(0x10011, 20);
                    break;
                case G1_CANALMAIS:
                    irsend.sendRC6(0x4C, 20);
                    break;
                case G1_CANALMENOS:
                    irsend.sendRC6(0x4D, 20);
                    break;
            }
        }
    }
}

```

```

    case G1_CANAL_1:
irsend.sendRC6(0x5, 20);
        break;
    case G1_CANAL_2:
irsend.sendRC6(0x4, 20);
        break;
    case G1_VOLTAR:
        group = GROUP_6;
        break;
    }
    break;
case GROUP_2:
    switch (idx)
    {
    case G2_LIGARSOM:
irsend.sendRaw(LIGARSOM2, 67, 32);
digitalWrite(48, HIGH);
        break;
    case G2_DESLIGARSOM:
irsend.sendRaw(DESLIGARSOM2, 67, 32);
digitalWrite(48, LOW);
        break;
    case G2_AUMENTARSOM:
irsend.sendRaw(AUMENTARSOM2, 67, 32);
        break;
    case G2_DIMINUIRSOM:
irsend.sendRaw(DIMINUIRSOM2, 67, 32);
        break;
    case G2_ESTACAOMAI:
irsend.sendRaw(ESTACAOMAI2, 67, 32);
        break;
    case G2_ESTACAOMENOS:
irsend.sendRaw(ESTACAOMENOS2, 67, 32);
        break;
    case G2_ESTACAOGRAVADA:
irsend.sendRaw(ESTACAOGRAVADA2, 67, 32);
        break;
    case G2_VOLTAR:
        group = GROUP_6;
        break;
    }
    break;
case GROUP_3:
    switch (idx)
    {
    case G3_LIGARAR:
irsend.sendRaw(LIGARAR2, 199, 32);
digitalWrite(47, HIGH);
        break;
    case G3_DESLIGARAR:

```

```

    irsend.sendRaw(DESLIGARAR2, 199, 32);
    digitalWrite(47, LOW);
    break;
case G3_VOLTAR:
    group = GROUP_6;
    break;
}
break;
case GROUP_4:
    switch (idx)
    {
    case G4_LIGARLUZ:
        digitalWrite(44, HIGH);
        digitalWrite(46, HIGH);
        break;
    case G4_DESLIGARLUZ:
        digitalWrite(44,LOW);
        digitalWrite(46,LOW);
        break;
    case G4_VOLTAR:
        group = GROUP_6;
        break;
    }
    break;
case GROUP_5:
    switch (idx)
    {
    case G5_ABRIRPORTA:
        if (T == 0){
            fechadura.step(-650);
            digitalWrite(45,LOW);
            T = 1;
            break;
        }
        else {
            break;
        }
    case G5_TRANCARPORTA:
        if (T == 1){
            fechadura.step(650);
            digitalWrite(45,HIGH);
            T = 0;
            break;
        }
        else {
            break;
        }
    case G5_VOLTAR:
        group = GROUP_6;
        break;

```

```
    }  
    break;  
case GROUP_6:  
    switch (idx)  
    {  
    case G6_TV:  
        group = GROUP_1;  
        break;  
    case G6_SOM:  
        group = GROUP_2;  
        break;  
    case G6_ARCONDICIONADO:  
        group = GROUP_3;  
        break;  
    case G6_LAMPADA:  
        group = GROUP_4;  
        break;  
    case G6_FECHADURA:  
        group = GROUP_5;  
        break;  
    }  
    break;  
}  
}
```

Referências

- ARDUINO. *Libraries*. 2012. Disponível em: <<https://www.arduino.cc/en/Reference/Libraries>>. Citado na página 30.
- ARDUINO. *Shields*. 2012. Disponível em: <<https://www.arduino.cc/en/Main/ArduinoShields>>. Citado na página 31.
- BASALLA, G. *La evolución de la tecnología*. Editorial Crítica, 2011. (Crítica/Historia y Teoría). ISBN 9788498921861. Disponível em: <<https://books.google.com.br/books?id=xxHVXwAACAAJ>>. Citado na página 20.
- BRAGA, N. *Tudo sobre relé*. 2010. Disponível em: <<http://www.newtoncbraga.com.br/index.php/como-funciona/597-como-funcionam-os-reles>>. Citado na página 26.
- BURATTINI, M. D. C. *Energia uma abordagem multidisciplinar*. LIVRARIA DA FÍSICA, 2008. ISBN 9788588325999. Disponível em: <<https://books.google.com.br/books?id=e71C1bInDocC>>. Citado na página 32.
- CAVALCANTE, M. A.; RODRIGUES, T. T. T.; BUENO, D. A. Controle remoto: princípio de funcionamento (parte 1 de 2). *Caderno Brasileiro de Ensino de Física*, v. 30, n. 3, p. 554–565, 2013. Citado na página 24.
- CONDIT, R.; JONES, D. W. Stepping motors fundamentals. *Microchip Application Note: AN907*, [Online]. Available: www.microchip.com, 2004. Citado na página 27.
- EDITORA, S. *Sistemas elétricos prediais: Instalação*. SENAI-SP Editora, 2015. (Eletroeletrônica). ISBN 9788583933038. Disponível em: <<https://books.google.com.br/books?id=kW8UDAAAQBAJ>>. Citado na página 25.
- EISMIN, T. *Eletrônica de Aeronaves: Introdução aos Sistemas Aviônicos*. [s.n.], 2016. ISBN 9788582604069. Disponível em: <<https://books.google.com.br/books?id=k1SnDAAAQBAJ>>. Citado na página 34.
- FRENZEL, L. *Eletrônica Moderna*. [s.n.], 2015. ISBN 9788580555363. Disponível em: <<https://books.google.com.br/books?id=MHDOCgAAQBAJ>>. Citado na página 26.
- IBGE, I. B. de Geografia e E. *Censo Demográfico 2010: Pessoas com deficiência*. Instituto Brasileiro de Geografia e Estatística - IBGE, 2010. Disponível em: <http://www.ibge.gov.br/estadosat/temas.php?sigla=rj&tema=censodemog2010_defic>. Citado na página 15.
- JOBSTRAIBIZER, F. *Desvendando as redes sem fio*. Universo dos Livros Editora, 2010. ISBN 9788578731021. Disponível em: <<https://books.google.com.br/books?id=OOER73iRfgkC>>. Citado na página 22.
- LAMB, F. *Automação Industrial na Prática - Série Tekne*. AMGH Editora, 2015. (Tekne). ISBN 9788580555141. Disponível em: <<https://books.google.com.br/books?id=X5XzCQAAQBAJ>>. Citado na página 27.
- LOMBARDI, R. R. R. Controle remoto infravermelho para automação. 2006. Citado na página 23.

- LUIZ, M. *A história da rede sem fio*. 2013. Disponível em: <<http://www.lifestyles.com.br/index.htm/2013/02/a-historia-da-rede-sem-fio/>>. Citado na página 21.
- MCRBERTS, M. *Arduino básico*. Editora Novatec, v. 344755160, 2011. Citado na página 29.
- PEREIRA, G. *Projeto de usinas hidrelétricas: passo a passo*. Oficina de Textos, 2015. ISBN 9788579751738. Disponível em: <<https://books.google.com.br/books?id=8HS3CQAAQBAJ>>. Citado na página 31.
- PIMENTA, A.; SALVADO, A. Deficiência e desigualdades sociais. *Sociedade e Trabalho*, v. 41, p. 155–166, 2010. Citado na página 15.
- PINTO, J. *Conversao Eletromecanica De Energia*. BIBLIOTECA 24 HORAS, 2011. ISBN 9788541601351. Disponível em: <<https://books.google.com.br/books?id=LWxdNQEACAAJ>>. Citado na página 31.
- RAMAMURTHY, B.; BHARGAVI, S.; SHASHIKUMAR, R. Development of a low-cost gsm sms-based humidity remote monitoring and control system for industrial applications. *International Journal of Advanced Computer Science and Applications*, Citeseer, v. 1, n. 4, 2010. Citado na página 27.
- SANTOS, L. Sistema de comunicação usb com microcontrolador. *Monografia para Graduação em Engenharia da Computação, UPE. Pernambuco*, 2009. Citado na página 29.
- SARTORETTO, M.; BERSCH, R. *Assistiva Tecnologia e Educação*. 2014. Disponível em: <<http://www.assistiva.com.br/tassistiva.html>>. Citado na página 15.
- ZHANG, T. *Instrumentation, Measurement, Circuits and Systems*. Springer Berlin Heidelberg, 2012. (Advances in Intelligent and Soft Computing). ISBN 9783642273346. Disponível em: <https://books.google.com.br/books?id=aLFdsFs3E_YC>. Citado na página 33.