



# **SISTEMA EMBARCADO VESTÍVEL E INTELIGENTE PARA AUXILIAR A LOCOMOÇÃO DE DEFICIENTES VISUAIS**

**Trabalho de Conclusão de Curso**

**Engenharia da Computação**

**Jonas Cordeiro do Nascimento Neto**  
**Orientador: Prof. Mêuser Jorge Silva Valença**



UNIVERSIDADE  
DE PERNAMBUCO

**Universidade de Pernambuco  
Escola Politécnica de Pernambuco  
Graduação em Engenharia de Computação**

**JONAS CORDEIRO DO NASCIMENTO NETO**

**SISTEMA EMBARCADO VESTÍVEL E  
INTELIGENTE PARA AUXILIAR A  
LOCOMOÇÃO DE DEFICIENTES  
VISUAIS**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

**Recife, Julho 2016.**

## MONOGRAFIA DE FINAL DE CURSO

### Avaliação Final (para o presidente da banca)\*

No dia 30 de junho de 2016, às 9:00 horas, reuniu-se para deliberar a defesa da monografia de conclusão de curso do discente JONAS CORDEIRO DO NASCIMENTO NETO, orientado pelo professor Mêuser Jorge Silva Valença, sob título Sistema embarcável vestível e inteligente para auxiliar a locomoção de deficientes visuais, a banca composta pelos professores:

**Sérgio Mario Lins Galdino**

**Mêuser Jorge Silva Valença**

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada       Aprovada com Restrições\*       Reprovada

e foi-lhe atribuída nota: 10,0 ( dez )

\*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O discente terá \_\_\_\_\_ dias para entrega da versão final da monografia a contar da data deste documento.

SÉRGIO MARIO LINS GALDINO

MÊUSER JORGE SILVA VALENÇA

*Dedico essa monografia a minha família, especialmente ao meu avô Jonas Cordeiro,  
que eu tenho o maior orgulho de carregar o seu nome.*

# Agradecimentos

Primeiramente, a Deus por ter tornado o sonho de fazer uma graduação possível.

A esta Universidade, seu corpo docente, especialmente aos professores Anna Lucia, Bruno, Campello, Carmelo, Edison, Gustavo, José Paulo, Josineide, Madeiro, e seu corpo administrativo, especialmente a Darlan e Diva.

Ao meu orientador Mêuser Valença, pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos.

A minha família, pelo amor, incentivo e exemplos que sempre me deram.

A minha namorada Cauanne Linhares e sua família por todo apoio e torcida.

Aos meus colegas que ajudaram a produzir o trabalho, especialmente o grande Gemerson Gerardo.

A galera antiga do Carranca22, especialmente o grande Moura

A equipe do GSE e de toda a Celpe que sempre me apoiaram quando precisei.

A galera do batutinhas.

Aos veteranos do curso de Engenharia da Computação.

A banda Racionais MC's pelas suas letras inspiradoras.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu mais sincero agradecimento.

*“Os verdadeiros vencedores não são apenas aqueles que chegam ao topo, porém aqueles que vieram de mais longe pelas estradas mais duras. Talvez a sua vitória nunca chegue às manchetes dos jornais. Mas, você conhece o significado delas, e é isso que conta”*

*(Ernest A. Fitzgerald)*

# Resumo

Diante do grande número de deficientes visuais e da falta de acessibilidade nas cidades, é cada vez mais necessário, além de outras medidas, o estudo de tecnologias assistivas, com a finalidade de possibilitar a esses deficientes, a execução de suas tarefas de maneira autônoma. Esse trabalho propôs o desenvolvimento de um dispositivo vestível, controlado por uma rede neural, que capte obstáculos e desníveis próximos ao deficientes visuais, utilizando sensores ultrassônicos dispostos em várias partes do corpo, que consiga notificar o portador do dispositivo, através de atuadores vibratórios. O sistema desenvolvido, mostrou-se bastante eficiente para identificação e notificação de obstáculos laterais, obstáculos abaixo da linha da cintura e pequenos desníveis ao redor, entretanto, necessita de aprimoramento para identificação de batentes e grandes desníveis no terreno.

**Palavras-chave:** Tecnologia assistiva, sistema embarcado, sensor ultrassônico

# Abstract

Facing the large number of visually impaired and the cities lack of accessibility, is increasingly necessary, in addition to other measures, the study of assistive technologies, in order to enable these disabled people, the execution of their tasks autonomously. This work proposes the development of a wearable device controlled by a neural network, which captures obstacles and ground irregularities near the visually impaired, using ultrasonic sensors disposed in various parts of the body, that can notify the device carrier, through vibration actuators. The system developed, proved to be very efficient for identification and notification of side obstacles, obstacles below the waist and small gaps around, however, needs improvement to identify stops and large unevenness on the ground.

**Keywords:** assistive technology , embedded system , ultrasonic sensor



# Sumário

<b>Capítulo 1 Introdução</b>	<b>1</b>
1.1 Motivação .....	1
1.2 Objetivos .....	2
1.2.1 Objetivo Geral.....	2
1.2.2 Objetivos Especificos.....	2
1.3 Estrutura da Monografia .....	3
<b>Capítulo 2 Fundamentação teórica</b>	<b>4</b>
2.1 Deficiência visual.....	4
2.2 Tecnologia assistiva .....	5
2.3 Pulse With Modulation (PWM).....	7
2.4 Sistemas Embarcados.....	9
2.4.1 Arduino .....	11
2.5 Sensores Ultrassônicos.....	12
2.5.1 HC-SR04 .....	14
2.6 Redes Neurais Artificiais .....	14
2.6.1 Neurônio biológico x Neurônio artificial.....	16
2.6.2 Treinamento da rede .....	16
2.6.3 Multi-Layer Perceptron (MLP).....	17
<b>Capítulo 3 Estado da arte</b>	<b>19</b>
3.1 Bengala com sensor ultrassônico.....	19

3.2	Pulseira com sensor ultrassônico .....	20
3.3	Óculos com sensor ultrassônico.....	21
3.4	Chapéu com sensor ultrassônico .....	22
<b>Capítulo 4 Materiais e Métodos</b>		<b>24</b>
4.1	Detecção de obstáculos com sensores ultrassônicos .....	24
4.2	Detecção de desníveis com sensores ultrassônicos .....	25
4.3	Descrição do dispositivo.....	26
4.4	Controle de múltiplos sensores ultrassônicos.....	27
4.5	Funcionamento do projeto .....	29
<b>Capítulo 5 Experimentos</b>		<b>30</b>
5.1	Cenário A .....	31
5.2	Cenário B .....	32
5.3	Cenário C .....	33
5.4	Cenário D .....	34
5.5	Cenário E .....	35
5.6	Cenário F.....	36
<b>Capítulo 6 Resultados obtidos</b>		<b>37</b>
<b>Capítulo 7 Considerações finais</b>		<b>40</b>
7.1	Conclusão .....	40
7.2	Trabalhos futuros.....	41
<b>Bibliografia</b>		<b>42</b>
<b>Anexo A Datasheet Arduino Mega</b>		<b>45</b>

<b>Apêndice B Diagrama elétrico</b>	<b>46</b>
<b>Apêndice C Código Fonte da Rede Neural</b>	<b>47</b>
<b>Apêndice D Exemplo dados utilizados para treinamento</b>	<b>52</b>
<b>Apêndice E Código Fonte do Arduino</b>	<b>53</b>

# Índice de Figuras

<b>Figura 1.</b>	Exemplo de variação no <i>duty cycle</i> .....	8
<b>Figura 2.</b>	PWM com <i>duty cycle</i> = 40 $\mu$ s .....	9
<b>Figura 3.</b>	Exemplo de modelo de subdivisão de um sistema embarcado .....	10
<b>Figura 4.</b>	Arduino Mega .....	12
<b>Figura 5.</b>	Funcionamento sensor ultrassônico .....	12
<b>Figura 6.</b>	Sensor Ultrassônico HC-SR04 .....	14
<b>Figura 7.</b>	Neurônio artificial McCulloch e Pitts .....	16
<b>Figura 8.</b>	Representação gráfica de uma RNA MLP com duas camadas intermediárias .....	18
<b>Figura 9.</b>	Bengala eletrônica .....	19
<b>Figura 10.</b>	Dispositivo Argos .....	20
<b>Figura 11.</b>	Usuário de bengala com o ARGOS .....	21
<b>Figura 12.</b>	Dispositivo <i>AnnuItWalk</i> .....	22
<b>Figura 13.</b>	Chapéu com sensor ultrassônico .....	23
<b>Figura 14.</b>	Módulo detector e notificador de obstáculos .....	25
<b>Figura 15.</b>	Módulo detector e notificador de desniveis .....	26
<b>Figura 16.</b>	Modelo do protótipo e protótipo construído .....	27
<b>Figura 17.</b>	Arquitetura da rede neural MLP usada .....	28
<b>Figura 18.</b>	Fluxograma da lógica de funcionamento do dispositivo .....	29
<b>Figura 19.</b>	Cenário de teste para sensores dos braços .....	31

<b>Figura 20.</b>	Cenário de teste para sensor da cabeça.....	32
<b>Figura 21.</b>	Cenário de teste para sensores das pernas .....	33
<b>Figura 22.</b>	Primeiro cenário de teste para sensores detectores de desniveis.....	34
<b>Figura 23.</b>	Segundo cenário de teste para sensores detectores de desniveis.....	35
<b>Figura 24.</b>	Terceiro cenário de teste para sensores detectores de desniveis .....	36
<b>Figura 25.</b>	Gráfico comparativo dos testes efetuados .....	39

# Índice de Tabelas

<b>Tabela 1.</b>	Resultados amostra sensor sem calculo de média .....	24
<b>Tabela 2.</b>	Resultados amostra sensor com calculo de média .....	25
<b>Tabela 3.</b>	Resultados Pessoa A .....	37
<b>Tabela 4.</b>	Resultados Pessoa B .....	38
<b>Tabela 5.</b>	Resultados Pessoa C .....	38

# Tabela de Símbolos e Siglas

CAT - Comitê de Ajudas Técnicas

cm – Centímetros

EEPROM - Electrically-Erasable Programmable Read-Only Memory

FPGA - Field Programmable Gate Array

IBGE - Instituto Brasileiro de Geografia e Estatística

ICD-9-CM - International Classification of Diseases, Clinical Modification

kB – KiloByte

kHz - KiloHertz

MHz - MegaHertz

MIT - Massachusetts Institute of Technology

MLP – Multi-Layer Perceptron

OMS- Organização Mundial de Saúde

PWM – Pulse With Modulation

RNA – Redes Neurais Artificiais

s - Segundos

SEDH/PR - Secretaria Especial dos Direitos Humanos da Presidência da República

SRAM – Static Random Access Memory

UART - Universal asynchronous receiver/transmitter

USB - Universal Serial Bus

UPE – Universidade de Pernambuco

V - Volts



# Capítulo 1

## Introdução

Este capítulo encontra-se dividido em três seções. Em seu início é descrito a motivação para o desenvolvimento deste trabalho. Em seguida, são discutidos seus principais objetivos. Por fim, é apresentada a estrutura dos próximos capítulos da monografia.

### 1.1 Motivação

Segundo o IBGE (Instituto Brasileiro de Geografia e Estatística), no censo de 2010[17], cerca de 45,6 milhões de brasileiros declararam ter alguma deficiência. A deficiência visual foi a que mais apareceu dentre as respostas dos entrevistados e chegou a 35,7 milhões de pessoas, entre as pessoas que declararam ter deficiência visual, mais de 6,5 milhões disseram ter a dificuldade de forma severa e 6 milhões afirmaram que tinham dificuldade de enxergar. Mais de 506 mil informaram serem cegas. No que tange a deficiência visual, a Organização Mundial de Saúde (OMS) estima que existem 39 milhões de cegos no mundo e outros 246 milhões sofrem de perda moderada ou severa da visão[20].

A falta de acessibilidade nas ruas e espaços públicos na maioria das cidades brasileiras afeta negativamente a vida de pessoas deficientes em seu cotidiano, acessibilidade é um ponto chave para se promover a integração social do indivíduo. Além disso, os deficientes visuais geralmente enfrentam vários riscos em sua locomoção pelas ruas, riscos estes ligados a possibilidade de queda e colisão com objetos fixos e móveis[16].

Para auxiliar sua locomoção, os deficientes contam com a ajuda da bengala que funciona como uma ferramenta de tato, conferindo se o espaço a sua frente é apto a ser percorrido, podendo detectar obstáculos que impeçam a passagem do deficiente, no entanto, essa ferramenta se mostra falha em casos de obstáculos acima da linha da cintura, e nesses casos em que a bengala falha, diversos

acidentes podem acontecer, levando o deficiente a se machucar e ainda a sofrer um efeito de trauma atrelado ao simples fato de andar sozinho na rua.

Existem diversos estudos que contemplam o desenvolvimento de tecnologias assistivas para complementar ou amplificar a utilização da bengala tais como: óculos[1], chapéus[11], pulseiras[24] e bengalas[8] equipados com sensores ultrassônicos, adaptações de residência[25], além de câmeras estereoscópicas[9].

Porém, esses estudos utilizam apenas um ou dois sensores ultrassônicos, o que diminui a quantidade de informações identificadas sobre obstáculos presentes no cenário, e no caso de uma falha na medição e/ou notificação, o portador do dispositivo pode vir a se machucar.

## **1.2 Objetivos**

### **1.2.1 Objetivo Geral**

Esse trabalho teve como objetivo geral, o desenvolvimento de um dispositivo, utilizando um sistema embarcado equipado com sensores ultrassônicos espalhados por várias partes do corpo, apto a identificar e alertar o deficiente visual de possíveis obstáculos e desníveis no relevo em seu caminho, através de atuadores vibratórios acionados com intensidade variável.

### **1.2.2 Objetivos Específicos**

Como objetivos específicos, esse trabalho se dispôs a:

- Discutir trabalhos presentes na literatura que tenham como objetivo, a construção de um sistema embarcado para auxiliar a locomoção de deficientes visuais;
- Investigar o uso de sensores ultrassônicos para detecção de obstáculos;

- Construir um protótipo vestível para auxiliar a locomoção de deficientes com a utilização de Arduino, sensores ultrassônicos e atuadores vibratórios;
- Utilizar uma rede neural para controlar a coleta de informações pelos sensores;
- Realizar testes de uso do dispositivo proposto em cenários reais;

## 1.3 Estrutura da Monografia

Essa monografia está dividida em 6 capítulos. No capítulo 2 é discutida a fundamentação teórica acerca de tecnologias assistivas, sistemas embarcados, sensores ultrassônicos, PWM (Pulse With Modulation ) e Redes Neurais. No capítulo 3, são apresentados trabalhos relacionados presentes no estado da arte de tecnologias assistivas para auxiliar a locomoção de deficientes visuais. No capítulo 4, é apresentado o modelo de dispositivo proposto, bem como suas especificações. Em seguida, no capítulo 5 são apresentados os resultados para o teste do dispositivo em campo. Finalizando com o capítulo 6, onde são concluídas as informações sobre esse trabalho e definidos possíveis trabalhos futuros.

# Capítulo 2

## Fundamentação teórica

Este capítulo apresenta conceitos acerca da base teórica sobre a qual o trabalho foi desenvolvido. As seções 2.1 e 2.2 tratam de conceitos base sobre deficiência visual e tecnologia assistiva. Na seção 2.3 é abordada a teoria sobre a modulação por largura de pulso (PWM), técnica utilizada para acionamento do atuador vibratório que vai notificar o deficiente visual sobre obstáculos em seu caminho. As seções 2.4 e 2.5 trazem o conceito de sistemas embarcados e sensores ultrassônicos, apresentando a tecnologia utilizada no estudo. Por fim, a seção 2.6 aborda fundamentos e a arquitetura da rede neural empregada neste trabalho.

### 2.1 Deficiência visual

Deficiência visual pode ser definida como a perda ou redução da capacidade visual em ambos os olhos em caráter definitivo, que não possa ser melhorada ou corrigida com o uso de lentes, tratamento clínico ou cirúrgico [15], isto é, não são considerados deficientes visuais, pessoas com doenças como miopia, astigmatismo ou hipermetropia, que podem ser corrigidas com o uso de lentes ou em cirurgias

Entende-se por acuidade visual o grau de aptidão do olho, para perceber a forma e o contorno dos objetos. Essa capacidade discriminatória é atributos dos cones (células fotossensíveis da retina), que são responsáveis pela acuidade visual, central que compreende a visão de forma e a visão de cores[6].

Segundo critérios estabelecidos pela Organização Mundial da Saúde (OMS) através da *ICD-9-CM*[27], os diferentes graus de acuidade visual e auxílios necessários podem ser classificados em:

- Visão normal: necessidade de lentes bifocais comuns;

- Próxima do normal: necessidade de bifocais mais fortes ou lupas de baixo poder;
- Baixa visão moderada: necessidade de lentes esfero prismáticas ou lupas mais fortes;
- Baixa visão severa: necessidade de lentes esféricas<sup>1</sup> ou lupas de mesa ou alto poder;
- Baixa visão profunda: necessidade de lupa montada, telescópio, bengala e/ou treinamento de orientação/mobilidade.
- Próxima à cegueira e cegueira total: livros falados, *braille*, aparelhos de saída de voz, bengala e/ou treinamento de orientação/mobilidade.

As causas mais frequentes que levam à cegueira infantil são doenças como: glaucoma congênito, a retinopatia da prematuridade, a rubéola, a catarata congênita, a toxoplasmose congênita, a hipovitaminose A, a oncocercose, o sarampo e o tracoma. Em relação a adultos, as causas mais frequentes são doenças como: catarata, diabetes, descolamento de retina, glaucoma, retinopatias além de causas acidentais[10].

## 2.2 Tecnologia assistiva

Tecnologia assistiva pode ser entendida como recursos e serviços que visam facilitar o desenvolvimento de atividades diárias por pessoas com deficiência. Procuram aumentar as capacidades funcionais e assim promover a independência e a autonomia de quem as utiliza[21]. De modo geral, pode-se simplificar esse termo como tecnologias que tem como objetivo principal o benefício de pessoas com deficiência, sendo tais tecnologias desenvolvidas com a função de auxiliar as habilidades dessas pessoas, com o intuito de lhes proporcionar uma maior autonomia e produtividade na vida.

---

<sup>1</sup>Lentes mais planas e finas do que as convencionais(esféricas)

Segundo Radabaugh(1993): “Para as pessoas sem deficiência a tecnologia torna as coisas mais fáceis. Para as pessoas com deficiência, a tecnologia torna as coisas possíveis”[23].

No Brasil, em novembro de 2006, através da portaria 142, foi instituído o Comitê de Ajudas Técnicas (CAT) pela Secretaria Especial dos Direitos Humanos da Presidência da República - SEDH/PR com objetivos principais de: 1)apresentar propostas de políticas governamentais e parcerias entre a sociedade civil e órgãos públicos referentes à área de tecnologia assistiva; 2)estruturar as diretrizes da área de conhecimento; 3)realizar levantamento dos recursos humanos que atualmente trabalham com o tema; 4)propor a criação de cursos na área de tecnologia assistiva, bem como o desenvolvimento de outras ações com o objetivo de formar recursos humanos qualificados e propor a elaboração de estudos e pesquisas, relacionados com o tema da tecnologia assistiva[7].

Os recursos de tecnologia assistiva são classificados de acordo com objetivos funcionais a que se destinam, dentre os quais podem ser citados:

- Auxílios para a vida diária e vida prática: materiais que favorecem o desempenho autônomo e independente em tarefas rotineiras ou facilitam o cuidado de pessoas em situação de dependência. Por exemplo, talheres para pessoas com mal de *Parkinson*, bengalas eletrônicas, roupas desenhadas com o intuito de facilitar a o vestir e o despir.
- Comunicação aumentativa e alternativa: materiais que diminuem a defasagem entre a necessidade de comunicação e a habilidade em falar e/ou escrever. Por exemplo, *softwares* que convertem texto em voz.
- Projetos arquitetônicos para acessibilidade: projetos estruturais que auxiliam a mobilidade. Por exemplo, rampas, adaptações em banheiros.

- Auxilidade de mobilidade: materiais para auxiliar a mobilidade. Por exemplo, muletas, bengalas, próteses, cadeiras de rodas manuais ou elétricas.
- Sistemas de controle de ambiente: dispositivos eletrônicos que permitem o controle do ambiente por pessoas com algum tipo de limitação. Por exemplo, acionadores de sistema de segurança por controle de voz.
- Auxílios para pessoas com surdez ou com déficit auditivos: materiais que diminuem a defasagem entre a necessidade de comunicação e problemas auditivos. Por exemplo, notificação vibratória, aparelho de surdez.

O desenvolvimento de tecnologias assistivas para deficientes visuais, tem como principal objetivo, permitir ao deficiente não sentir ou minimizar a falta de um dos seus principais sentidos em seu cotidiano: a visão. Essas tecnologias, podem variar de uma simples bengala a um complexo sistema computadorizado, podendo ser também brinquedos, roupas adaptadas, computadores, softwares e hardwares especiais, que contemplam questões de acessibilidade, recursos para mobilidade, equipamentos de comunicação alternativa, chaves e acionadores especiais, aparelhos de escuta assistida, auxílios visuais, entre outros.

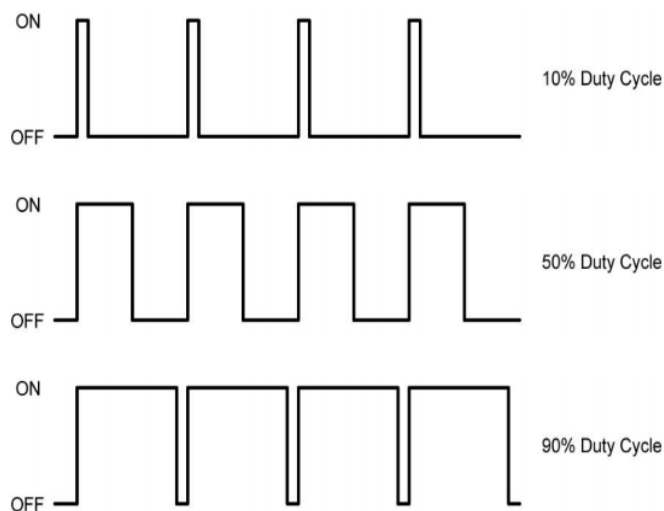
## 2.3 Pulse With Modulation (PWM)

PWM, do inglês Pulse Width Modulation, é uma técnica utilizada para variação do valor médio de uma forma de onda periódica. A técnica consiste em manter a frequência de uma onda quadrada fixa e variar o tempo que o sinal fica em nível lógico alto. Esse tempo é chamado de *duty cycle*, ou seja, o ciclo ativo da forma de onda. Na Figura 1 são exibidos alguns exemplos de variação no *duty cycle*[22].

A utilização de circuitos PWM como controladores ao invés de circuitos resistivos, tem como principal vantagem a eficiência. Enquanto o PWM trabalha com

eficiência de quase 100%, para um circuito resistivo trabalhando a 50% da carga, 50% vai realmente para alimentação da carga e 21% é perdido em aquecimento nos resistores.

**Figura 1.** Exemplo de variação no *duty cycle*



[Fonte: MecaWeb<sup>2</sup>]

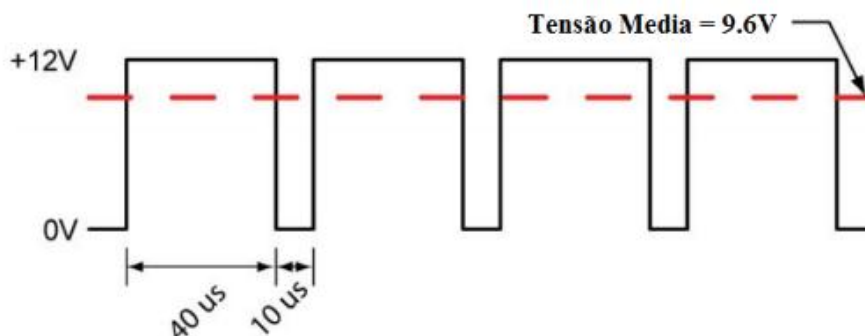
A tensão média de saída pode ser obtida com a partir da equação 1 e pode ser visualizada na Figura 2.

$$Tensão\ Média = Tensão\ Entrada * 100 * \frac{duty\ cycle}{periodo} \quad (1)$$

---

<sup>2</sup> Disponível em: [http://www.mecaweb.com.br/eletronica/content/e\\_pwm](http://www.mecaweb.com.br/eletronica/content/e_pwm) Acesso em 06 de Junho de 2016



Figura 2. PWM com *duty cycle* = 40  $\mu$ s[Fonte: MecaWeb<sup>2</sup>]

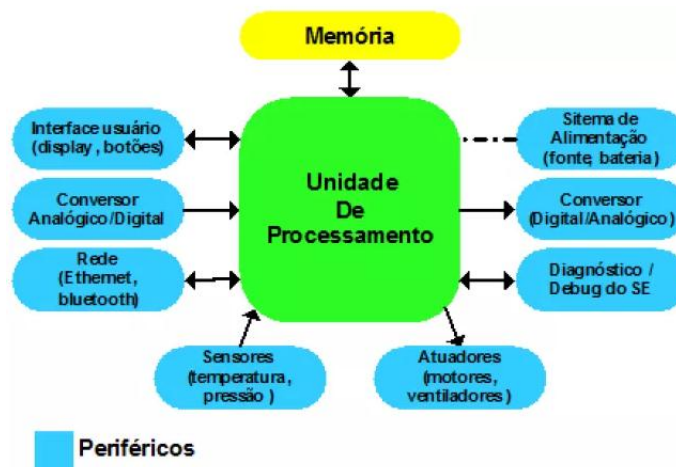
## 2.4 Sistemas Embarcados

Segundo Tanenbaum (2003), um sistema embarcado “é um sistema computacional fisicamente limitado, geralmente com restrições de memória, tamanho, energia e, consecutivamente, potência, que possui um número limitado, mas específico, de funções”. Geralmente está associado ou mesmo embutido em outro produto, como um eletrodoméstico ou um veículo. Muitas vezes possuem características de sistemas de tempo real, como alta velocidade de envio, tratamento e recebimento de dados.

Ball (2005) define sistemas embarcados como: “Um sistema dedicado a uma única tarefa e interage continuamente com o ambiente a sua volta por meio de sensores e atuadores”.

Em projetos de sistemas embarcados, normalmente pode-se subdividir o modelo em 3 partes: unidade de processamento, memória e periféricos, como pode ser visto na Figura 3.

Figura 3. Exemplo de modelo de subdivisão de um sistema embarcado



[Fonte: Blog de sistemas embarcados<sup>3</sup>]

A unidade de processamento executa as instruções de cálculo, desvio condicional e incondicional, tratamento de eventos, entre outras. Sistemas embarcados utilizam geralmente, para o processamento, microprocessadores, porém existem outras opções como microcontroladores ou processadores de sinal digital.

A memória armazena dados e instruções relacionados às operações da unidade de processamento, podendo ser divididas entre memória de dados e memória de instrução dependendo da arquitetura.

Os periféricos são as interfaces de entrada e saída da unidade de processamento com o mundo externo, atuadores interagem como dispositivos de saída e sensores atuam como dispositivos de entrada.

O primeiro sistema embarcado reconhecido mundialmente foi o *Apollo Guidance Computer*, desenvolvido por Charles Stark Draper no MIT, que era um computador de bordo que provia recursos computacionais e controles para

<sup>3</sup> Disponível em: <http://www.sistemasembarcados.org/artigo-sistemas-embarcados-o-que-e-definicao.html> Acesso em 06 de Junho de 2016

orientação, navegação e controle do módulo de comando e do módulo lunar utilizados no Projeto *Apollo*[13].

Sistemas embarcados estão cada vez mais baratos e por consequência mais acessíveis, demandando menor consumo de energia e, além da miniaturização, possuem poder de processamento cada vez maior. Com essa relação custo x benefício crescente, sistemas embarcados estão se tornando cada vez mais populares, contribuindo para um mundo cada vez mais micro conectado, onde não só os computadores compartilham informações, mas também os objetos ao nosso redor.

### 2.4.1 Arduino

Arduino é uma plataforma de prototipagem eletrônica *open source*, que tem sido cada vez mais utilizada, projetada geralmente com um microcontrolador Atmel AVR com suporte de entrada e saída embutidos, uma linguagem de programação padrão, a qual tem origem em Wiring, e incluem elementos das linguagens C/C++. Tem como principal objetivo do projeto a possibilidade de criação de ferramentas que são acessíveis, com baixo custo, flexíveis e fáceis de se usar[3].

O projeto iniciou-se na cidade de Ivrea, na Itália, em 2005, com o intuito de interagir em projetos escolares de forma a ter um orçamento menor que outros sistemas de prototipagem disponíveis naquela época. Atualmente existem cerca de 13 versões comerciais da placa.

O arduino Mega, versão utilizada nesse trabalho, é baseado no microcontrolador ATmega1280, que possui 54 pinos digitais de entrada e saída, onde 14 podem ser utilizadas como saída PWM, 16 entradas analógicas, 4 portas seriais UARTs, um cristal oscilador que fornece frequência de operação de 16MHz e uma conexão USB. Essa placa tem como tensão operacional 5V, com tensão de entrada recomendada de 7 a 12V, possui cerca de 128KB de memória *Flash*, utilizada para armazenar os programas, 8kB de memória SRAM e 4KB de memória EEPROM[2], e pode ser vista na Figura 4. O *datasheet* da Arduino Mega pode ser visto no anexo A.

Figura 4. Arduino Mega

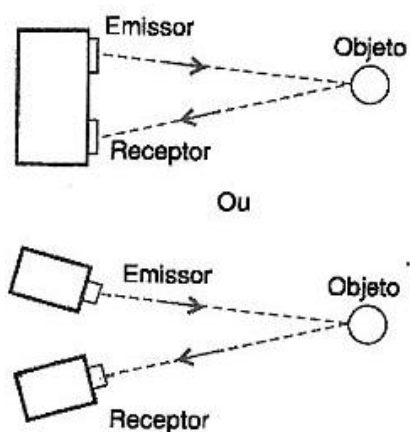


[Fonte: Pagina principal do Arduino Mega<sup>4</sup>]

## 2.5 Sensores Ultrassônicos

O princípio de funcionamento desses sensores, baseia-se em animais como o morcego, que emite ultrassons, e baseado na sua reflexão, são capazes de detectar objetos e presas em seu vôo cego, conforme pode ser visto na Figura 5.

Figura 5. Funcionamento sensor ultrassônico



[Fonte: Instituto Newton C. Braga<sup>5</sup>]

<sup>4</sup> Disponível em: <https://www.arduino.cc/en/Main/ArduinoBoardMega> Acesso em 06 de Junho de 2016

Esses sensores se caracterizam por operar através de um tipo de radiação não sujeita a interferência eletromagnética, tornando-os mais robustos para determinados tipos de aplicação.

Sensores ultrassônicos, estão se tornando cada vez mais populares na indústria, sendo aplicados nas mais diversas situações, tais como: detecção de passagem de objetos em uma linha de montagem, detecção da presença de pessoas ou ainda de substâncias em seus diversos estados num reservatório permitindo a medida de seu nível[12].

Ao utilizar sensores ultrassônicos, alguns fatores devem ser levados em consideração, pois dependendo da aplicação, podem afetar o bom desempenho do sistema, são eles:

- Reflexões não desejadas: dependendo do material a ser detectado, o ultrassom pode passar por reflexões múltiplas, ou o sinal pode penetrar o objeto, levando a um erro na distância real do objeto.
- Região mínima e máxima: deve-se ser respeitado os limites mínimos e máximos que o sensor consegue captar, de modo a ter uma medição mais próxima do real.
- Características direcionais: o objeto a ser detectado, deve estar disposto na região de maior intensidade do sinal emitido, geralmente algo em torno de 8 a 30 graus.
- Fenômeno da reverberação: o objeto que reflete o sinal, devido a características do seu material, pode provocar o prolongamento do sinal recebido durante sua reflexão, gerando um efeito que pode influenciar a informação de distância.

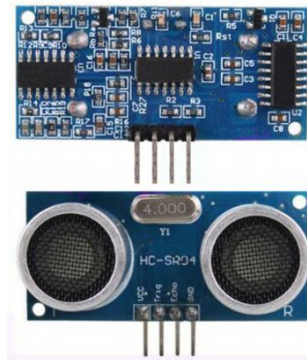
---

<sup>5</sup> Disponível em: <http://www.newtonbraga.com.br/index.php/como-funciona/5273-art691>  
Acesso em 06 de Junho de 2016

### 2.5.1 HC-SR04

O HC-SR04 conforme Figura 6, é um sensor ultrassônico que utiliza um sonar para detectar objetos. Ele é composto de um transmissor ultrassônico e um receptor ultrassônico que funcionam a 40kHz, capaz de detectar objetos no intervalo de 2cm à 400cm, com ângulo de abertura de 30°, não sendo afetado por raios de sol ou materiais com de tonalidade escura[14].

**Figura 6.** Sensor Ultrassônico HC-SR04



[Fonte: Robocore<sup>6</sup>]

Para a detecção do objeto, o pino Trig do sensor deve receber um pulso alto de 5V por pelo menos 10µs, ao receber o sinal refletido pelo obstáculo, o pino Echo vai ser ativado por um período de tempo ( $T$ ). Para obter a distância em centímetros, basta efetuar o cálculo de acordo com a equação 2.

$$Distância (cm) = \frac{T}{58} \quad (2)$$

## 2.6 Redes Neurais Artificiais

Redes Neurais Artificiais são modelos computacionais, ainda utilizados nos últimos anos, que tem seu funcionamento fortemente baseado no sistema nervoso

---

<sup>6</sup> Disponível em: [https://www.robocore.net/modules.php?name=GR\\_LojaVirtual&prod=620](https://www.robocore.net/modules.php?name=GR_LojaVirtual&prod=620)  
Acesso em 06 de Junho de 2016

central, capazes de aprender a partir de um conjunto de informações apresentadas, processo este chamado de treinamento da rede.

As RNAs são formadas por neurônios artificiais interligados através de conexões sinápticas, e cada conexão entre dois neurônios está associada a um peso sináptico[26].

Dentre as principais áreas de aplicações de RNAs podem ser citadas[19]:

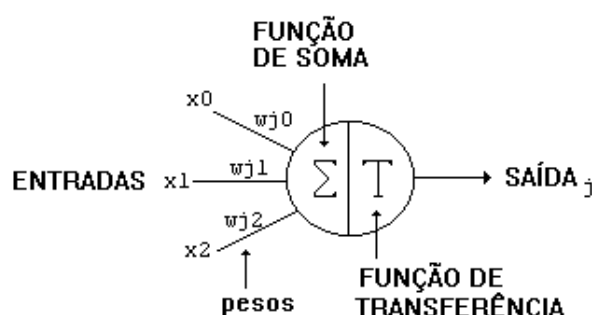
- Aproximação de funções: RNAs são treinadas com o objetivo de obter uma função aproximada que mapeie o relacionamento entre entradas e saídas apresentadas a rede;
- Classificação: RNAs são treinadas com o objetivo de associar cada padrão de entrada a uma classe definida;
- Controle de processos: RNAs são treinadas com o objetivo de identificar ações de controle que permitam o alcance do bom funcionamento de um sistema dinâmico;
- Otimização de sistemas: RNAs são treinadas com o objetivo de minimizar ou maximizar uma função, obedecendo a critérios relacionados ao domínio da aplicação;
- Previsão de series temporais: RNAs são treinadas com o objetivo de estimar a informação de uma variável através de observações realizadas ao longo do tempo;
- Reconhecimento de padrões: RNAs são treinadas com o objetivo de descobrir padrões existentes nos dados de entrada;

Para utilização prática de uma rede neural, alguns aspectos devem ser levados em consideração, como a capacidade de generalização, isto é, a RNA deve fornecer um resultado satisfatório quando submetida a um conjunto de dados que não foi utilizado para seu treinamento[29].

### 2.6.1 Neurônio biológico x Neurônio artificial

O primeiro modelo de neurônio artificial, conforme Figura 7, foi proposto por Mc Culloch e Pitts em 1943, composto por um conjunto de entradas, também conhecido como estímulos de entrada, uma unidade de processamento e uma ou mais saídas, que correspondem aos dendritos, corpo celular e terminais axônicos do neurônio biológico.

**Figura 7.** Neurônio artificial McCulloch e Pitts



[Fonte: Cerebromente<sup>7</sup>]

O funcionamento do neurônio obedece a lei do “Tudo ou Nada”, que afirma que se a soma das intensidades ponderadas dos estímulos não atingirem o limiar excitatório mínimo, a informação não é passada a diante[29].

### 2.6.2 Treinamento da rede

O processo de treinamento mais comum da rede neural, consiste em apresentar exemplos para a rede e avaliar, através de uma métrica previamente estabelecida, a capacidade da rede neural produzir uma saída que se aproxime da saída desejada, após a repetição do processo por vários ciclos[29].

Dois tipos de treinamentos podem ser destacados:

---

<sup>7</sup> Disponível em: <http://www.cerebromente.org.br/n05/tecnologia/rna.htm> Acesso em 06 de Junho de 2016



- Supervisionado: onde as saídas desejadas são apresentadas juntamente com as entradas;
- Não-supervisionado: onde apenas as entradas são apresentadas a rede neural;

Para encerrar o treinamento da rede neural, algumas abordagens pode ser utilizada:

- Número de ciclos: são definidos  $N$  ciclos de treinamento;
- Erro tolerável: é definido um erro máximo tolerável;
- Erro de validação cruzada: são apresentados conjuntos de exemplos que não fizeram parte da treinamento da rede e em seguida é calculado o erro para esses exemplos, o treinamento é encerrado ao erro de validação cruzada atingir seu valor mínimo[29].

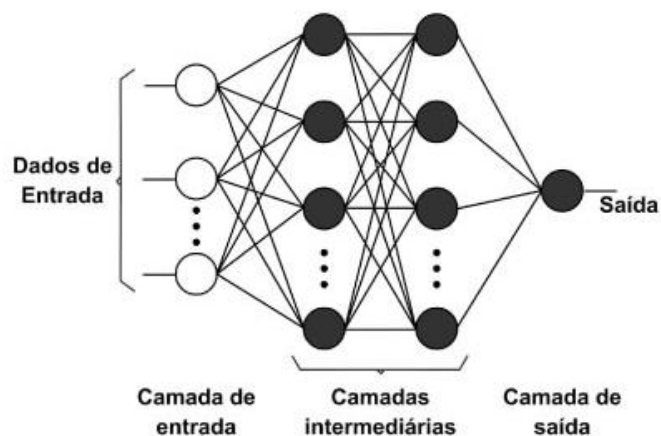
### 2.6.3 Multi-Layer Perceptron (MLP)

Os primeiros modelos de redes neurais foram o *Perceptron* e o *Adaline* propostos por Roseblatt em 1958 e Widrow e Hoff em 1960 respectivamente. O *Perceptron* trabalhava apenas com informações discretas, enquanto o *Adaline* permitia trabalhar com informações contínuas, porém, ambos possuem a limitação de funcionamento apenas para problemas linearmente separáveis[26].

As redes MLP são uma generalização da rede *Perceptron*, que possuem, além das camadas de entrada e saída, pelo menos uma camada intermediária, também conhecida como camada escondida. A adição dessa camada, permite a rede neural aproximar qualquer função não linear [29].

A Figura 8 exemplifica uma rede neural MLP que possui duas camadas intermediárias.

**Figura 8.** Representação gráfica de uma RNA MLP com duas camadas intermediárias



[Fonte: Scielo<sup>8</sup>]

O algoritmo mais utilizado para treinamento das redes MLP é o algoritmo *backpropagation*, proposto por Paul Werbos em 1970, que baseia-se no aprendizado por correção de erro retropropagado, isto é, o erro é retropropagado da camada de saída, para a(s) camada(s) intermediária(s), e posteriormente para a camada de entrada[29].

O algoritmo *backpropagation*, consiste de duas fases:

- Fase *forward*: propagação do estímulo de entrada até a saída e cálculo do erro;
- Fase *backward*: retropropagação do erro nas camadas anteriores, e reajuste dos pesos;

<sup>8</sup> Disponível em: [http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S1806-11172011000100009](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1806-11172011000100009) Acesso em 06 de Junho de 2016

# Capítulo 3

## Estado da arte

Este capítulo apresenta uma discussão sobre 5 soluções propostas na literatura, utilizando sistemas embarcados, com o objetivo de auxiliar a locomoção de deficientes visuais.

### 3.1 Bengala com sensor ultrassônico

Cardozo Bueno (2010) propôs o desenvolvimento de uma bengala, exibida na Figura 9, equipada com um sistema embarcado capaz de identificar e notificar deficientes visuais, sobre a presença de obstáculos acima e abaixo da linha da cintura.

**Figura 9.** Bengala eletrônica



[Fonte: Cardozo Bueno(2010)]

O dispositivo construído detecta obstáculos com distância entre 2 a 4 metros, e notifica o portador com sinais vibratórios e sinais sonoros e tem autonomia de 4 horas.

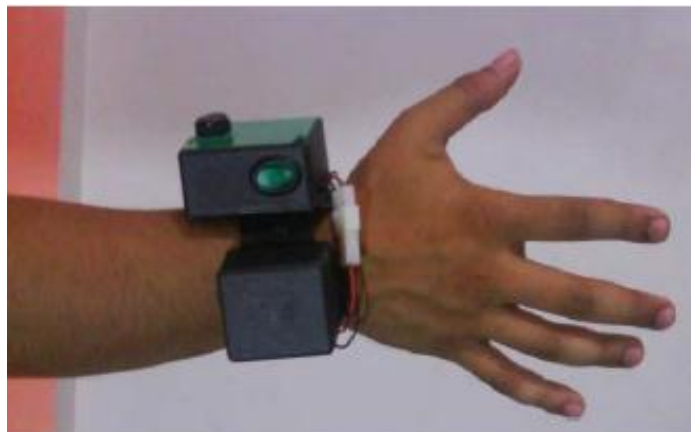
Algumas observações podem ser feitas em relação utilização da bengala eletrônica, como:

- Não detectar obstáculos a menos de 2 metros;
- Notificação com intensidade fixa, o que pode levar a uma não compreensão da distância real do obstáculo e se o mesmo esta se afastando ou se aproximando;
- Não detecta desníveis ou buracos;
- Aumento de peso da bengala;

### 3.2 Pulseira com sensor ultrassônico

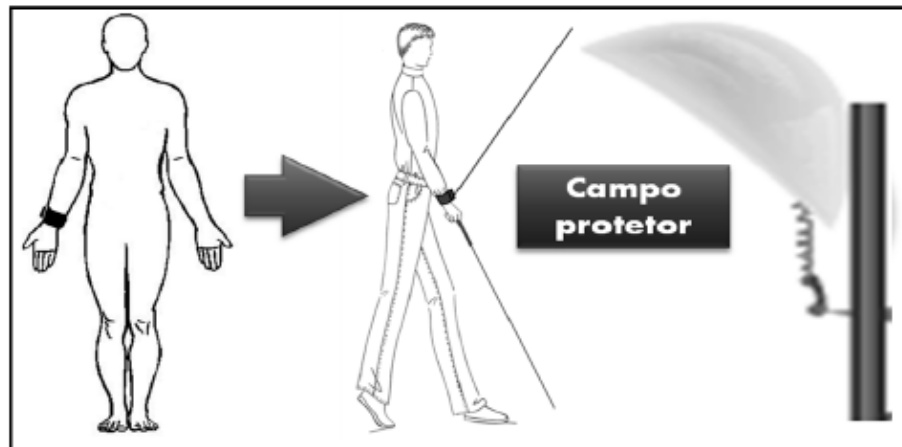
Com o objetivo de auxiliar a locomoção de deficientes visuais, Ribeiro(2011) propôs o dispositivo Argos, exibido na Figura 10, uma pulseira microcontrolada, equipada com um sensor ultrassônico para detecção de obstáculos e um atuador vibratório para notificar o portador do dispositivo.

**Figura 10.**Dispositivo Argos



[Fonte: Ribeiro(2011)]

O ARGOS, foi desenvolvido com o objetivo de ser utilizado em conjunto com a bengala branca, detectando obstáculos acima da linha da cintura, com proteção exibido na Figura 11.

**Figura 11.** Usuário de bengala com o ARGOS

[Fonte: Ribeiro(2011)]

O dispositivo possui dois modos de funcionamento, quando está operando no canal 1, emite um pulso vibratório alto ao detectar um obstáculo com até 1 metro de distância, caso esteja operando no canal 2, emite um pulso vibratório com intensidade variável ao detectar um obstáculo com até 2 metros de distância.

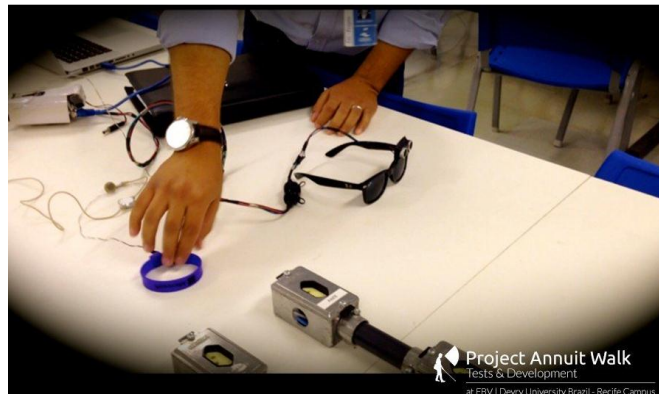
Um problema porém reside na utilização do ARGOS, como não substitui a bengala, o portador do dispositivo pode estar sujeito a quedas devido a desníveis ou buracos, além de colisões laterais, em virtude da utilização de apenas um sensor ultrassônico, que limita a quantidade de informações ao seu redor.

### 3.3 Óculos com sensor ultrassônico

A Startup pernambucana *WearIT* em 2014, desenvolveu um dispositivo chamado *AnnuitWalk*, um óculos equipado com um sensor ultrassônico, para, em conjunto com a bengala, auxiliar a locomoção de deficientes visuais.

Esse dispositivo, exibido na Figura 12, é capaz de detectar obstáculos a um ângulo de 120°, e notificar o portador através de uma pulseira que emite uma vibração, dependendo de quão próximo o obstáculo se encontra

**Figura 12.** Dispositivo *AnnuItWalk*



[Fonte: *AnnuItWalk*<sup>9</sup>]

O *AnnuItWalk* ainda pode ser utilizado em conjunto com um aplicativo, capaz de informar ao portador do dispositivo a melhor rota ou obstáculos próximos, baseado em informações coletadas e enviadas por outros usuários do mesmo aplicativo.

A utilização de apenas um sensor ultrassônico entretanto, limita a quantidade de objetos detectados no ambiente, e em caso de falha na medição, ou de múltiplos obstáculos, pode levar a acidentes.

### **3.4 Chapéu com sensor ultrassônico**

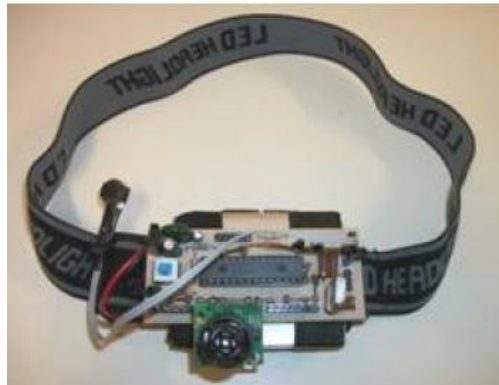
Para auxiliar a locomoção de deficientes visuais, Gonçalves(2010) propôs o desenvolvimento de um sistema embarcado, equipado com um sensor ultrassônico que ser utilizado na testa ou em um chapéu.

O dispositivo, exibido na Figura 13, identifica obstáculos acima da linha da cintura, com distâncias de até 2 metros, e notifica o deficiente visual através de um sinal vibratório com intensidade variável, dependendo da distância em que se encontra o obstáculo detectado.

---

<sup>9</sup> Disponível em: <http://annuitwalk.com/wordpress/2014/08/17/showing-the-paw-iii-prototype/>  
Acesso em 06 de Junho de 2016

**Figura 13.** Chapéu com sensor ultrassônico



[Fonte: Gonçalves(2010)]

Novamente pode-se comentar, que a utilização de apenas um sensor ultrassônico, limita a quantidade de objetos detectados no ambiente. Estereoscopia para detecção de obstáculos

Medeiros(2013) propôs a utilização de câmeras estereoscópicas e processamento de imagem, para detecção de objetos, e posteriormente, notificação do deficiente visual.

O sistema foi baseado no cálculo de correspondências e disparidades entre duas imagens estereoscópicas, utilizando a transformada Census e o cálculo de distância de Hamming, embarcados em um FPGA (*Field Programmable Gate Array*), para estimar a distância até os obstáculos.

Algumas observações entretanto, podem ser feitas sobre a utilização de processamento de imagem para detecção de obstáculos:

- Demanda um hardware e câmeras potentes, mais energia e grande tempo de processamento para cenários reais;
- Ruído, mudança na iluminação e desalinhamento das câmeras, podem impactar na detecção;
- Não detecta desníveis ou buracos;

# Capítulo 4

## Materiais e Métodos

Este capítulo apresenta como o trabalho foi desenvolvido e quais decisões foram tomadas durante o processo. As seções 4.1 e 4.2 descrevem como foram implementados os módulos de detecção de obstáculos e desníveis. Na seção 4.3, é apresentado o protótipo construído. A seção 4.4 traz a decisão tomada sobre o modo de controle de acionamento dos múltiplos sensores ultrassônicos utilizados. Por fim a seção 4.5 descreve a lógica de funcionamento do dispositivo.

### 4.1 Detecção de obstáculos com sensores ultrassônicos

Foi estudado o comportamento do sensor ultrassônico na detecção de objetos. Com isso o auxílio de uma régua, foi posicionado um objeto a 50cm do sensor e coletadas 30 amostras de leitura de distância, que estão descritas na Tabela 1.

**Tabela 1.** Resultados amostra sensor sem calculo de média

51	53	49	72	51	47	46	52	57	45	37	56	56	54	51
45	67	51	69	51	73	47	49	72	54	52	55	55	63	54

[Fonte: Autoria própria]

Conforme os dados obtidos, foi possível verificar que existia uma variação nos valores lidos que podem impactar no processo de notificação. Para o problema encontrado, foi efetuado o calculo de média móvel a cada 3 medições, conforme esperado, os dados obtidos ficaram mais concentrados na região próxima ao valor de 50cm e podem ser visualizados na Tabela 2.



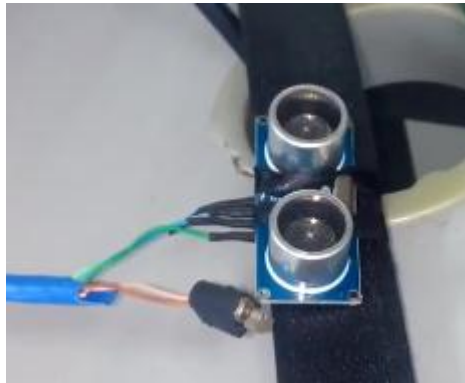
**Tabela 2.** Resultados amostra sensor com calculo de média

47	54	53	51	53	54	52	49	47	48	47	49	52	47	50
54	52	49	47	49	45	52	52	54	54	52	54	52	55	43

[Fonte: Aatoria própria]

Após o estudo do comportamento do sensor ultrassônico, foi construído um módulo, composto de um sensor ultrassônico e um atuador vibratório, para efetuar a detecção de obstáculos e notificação do deficiente visual. A Figura 14 exibe o módulo construído para a detecção e notificação de obstáculos.

**Figura 14.** Módulo detector e notificador de obstáculos



[Fonte: Aatoria própria]

Para notificação utilizando esse módulo, a seguinte regra foi estabelecida:

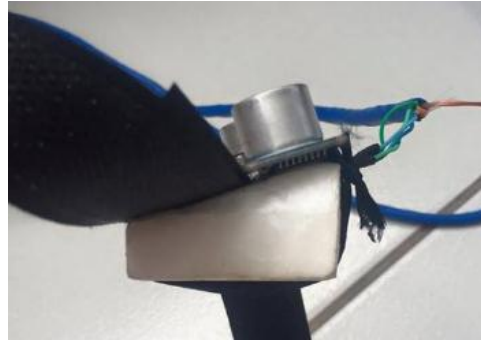
- Caso a distância lida esteja entre 5 *cm* e 180 *cm*, trata-se de um obstáculo e deve ser notificado com maior intensidade caso a distância se aproxime de 180 *cm*.

## 4.2 Detecção de desníveis com sensores ultrassônicos

Para detecção de desníveis e batentes, o módulo descrito na seção 4.1 foi alterado de modo a criar uma angulação de 30°, permitindo que a região de maior intensidade de reflexão do ultrassom, seja a área na qual o deficiente visual vai

percorrer no próximo passo. A Figura 15 exhibe o módulo construído para a detecção e notificação de desníveis.

**Figura 15.** Módulo detector e notificador de desníveis



[Fonte: Autoria própria]

Foram utilizadas duas métricas para notificação com esses módulos:

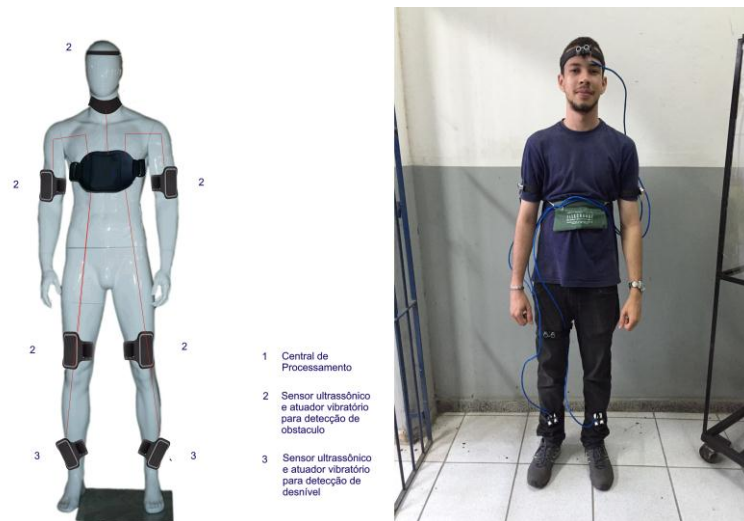
- Caso a distância lida esteja entre 60cm e 120cm, trata-se de um desnível e deve ser notificado com maior intensidade caso a distância se aproxime de 120cm.
- Caso a distância lida esteja entre 20cm e 5cm, trata-se de um batente e deve ser notificado com maior intensidade caso a distância se aproxime de 5cm.

### 4.3 Descrição do dispositivo

Para construção do dispositivo, em conjunto com o Arduino Mega, foram utilizados 4 módulos descritos na seção 4.1, dispostos nos braços, pernas e na cabeça do deficiente visual, e 2 módulos descritos na seção 4.2 próximo ao pé do portador do dispositivo, para detectar tanto obstáculos como desníveis, e notificar o portador do equipamento.

A Figura 16 demonstra um comparativo entre o modelo do protótipo idealizado e protótipo construído.

**Figura 16.** Modelo do protótipo e protótipo construído



[Fonte: Elaboração própria]

O dispositivo construído, é capaz de identificar obstáculos em múltiplas direções, com distâncias entre 10cm à 200cm, e notificar o portador do aparelho, com intensidade mais forte, caso o obstáculo se encontre mais próximo dele.

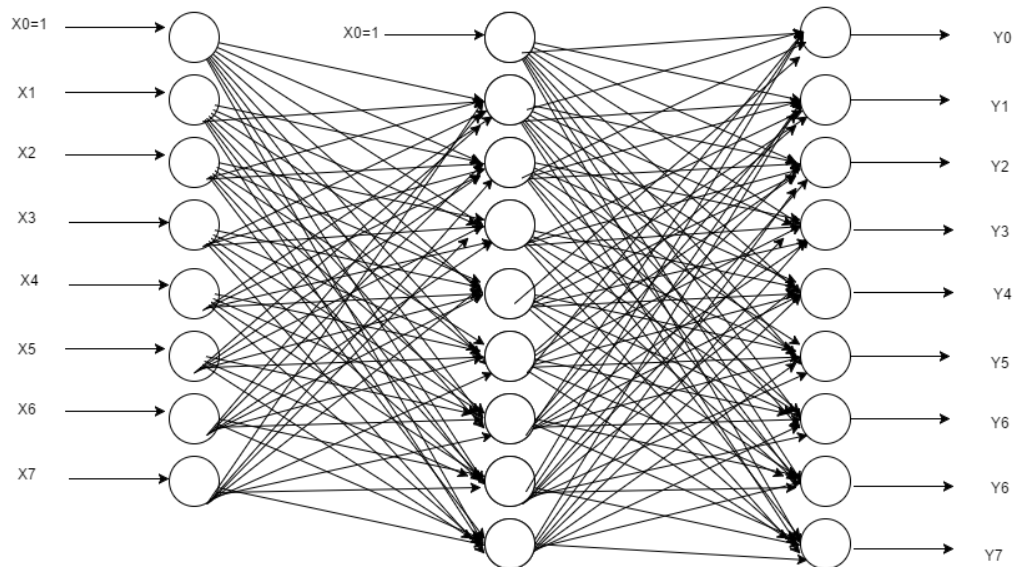
Para aumentar a confiabilidade do sistema, foi utilizado o módulo *watchdog* do Arduino, um circuito temporizador que emite um sinal de reset caso o programa principal pare de responder por mais do que 4 segundos.

## 4.4 Controle de múltiplos sensores ultrassônicos

Devido a utilização de múltiplos sensores ultrassônicos e da limitação de que a coleta de informação de distância não pode ser feita de forma paralela por todos os sensores, faz-se necessário o desenvolvimento de um controlador que possa priorizar a busca de informação de distância por um certo sensor S cuja medição no instante anterior tenha detectado um obstáculo muito próximo, caso contrário, o sensor S precisaria esperar que todos os sensores colem as informações de distância para então conseguir coletar a distância novamente, tendo a possibilidade de notificar a atualização da posição do obstáculo para o portador do dispositivo de maneira tardia .

Para solução desse problema, esse trabalho propôs a utilização do controlador utilizando uma rede neural MLP de modo a estimar a melhor combinação para acionamento dos sensores ultrassônicos. A Figura 17 demonstra a arquitetura da rede neural proposta.

**Figura 17.** Arquitetura da rede neural MLP usada



[Fonte: Autoria própria]

As entradas da rede neural  $x_1.. x_7$  correspondem respectivamente as leituras do: sensor da cabeça, sensor do braço esquerdo, sensor do braço direito, sensor da perna esquerda, sensor da perna direita, sensor do desnível esquerdo, sensor do desnível direito.

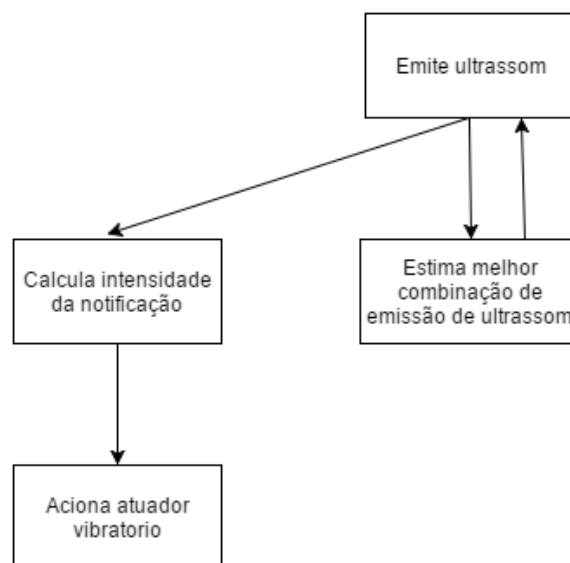
As saídas da rede neural  $y_0$  e  $y_3$  correspondem ao acionamento do sensor que detectou um obstáculo ou desnível mais próximo do portador do dispositivo, enquanto as saídas  $y_1..y_2$  e  $y_5..y_7$  correspondem ao acionamento dos sensores que detectaram obstáculos ou desníveis próximos até os que detectaram obstáculos mais distantes.

A rede neural foi treinada com o algoritmo *backpropagation* durante 100000 ciclos, utilizando um conjunto de exemplos parcialmente exemplificado no Apêndice D, e obteve um erro médio quadrático de validação cruzada de 11,965%

## 4.5 Funcionamento do projeto

O fluxograma exibido na Figura 18, demonstra a lógica de funcionamento do dispositivo proposto. Inicialmente, para cada sensor conectado ao sistema, é emitido um ultrassom e calculado o tempo para recepção do sinal refletido, obtendo-se assim a distância na qual o obstáculo mais próximo se encontra. Em seguida, é feito um cálculo baseado nessa distância para descobrir a intensidade na qual o portador do dispositivo deve ser notificado. Posteriormente, o atuador vibratório específico é acionado com a intensidade descoberta. Por fim, com as informações de distância lidas através de cada sensor, a rede neural estima a melhor combinação de emissão de ultrassom, de modo a priorizar medições em sensores que anteriormente captaram obstáculos próximo. O processo então é reinicializado.

**Figura 18.** Fluxograma da lógica de funcionamento do dispositivo



[Fonte: Autoria própria]

# Capítulo 5

## Experimentos

Para verificação do dispositivo proposto, foram selecionados alguns locais da cidade de Olinda, descritos nas seções 5.1 a 5.6, de modo a simular situações que o deficiente visual pode enfrentar em seu cotidiano.

Foram elaborados ainda, critérios de aceitação de modo a identificar se o dispositivo construído cumpriu os objetivos propostos em cada cenário e verificar possíveis pontos de melhoria.

## 5.1 Cenário A

O Cenário A apresentado pela Figura 19, tem por objetivo, representar situações onde o deficiente visual possa ter obstáculos em seu lado direito e/ou esquerdo, como por exemplo: muros, postes, entre outros.

Critérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um obstáculo a sua direita e/ou esquerda e caminhar sem encostar no muro.

**Figura 19.** Cenário de teste para sensores dos braços



[Fonte: Elaboração própria]

## 5.2 Cenário B

O Cenário B apresentado pela Figura 20, tem por objetivo, representar situações onde o deficiente visual possa ter obstáculos acima da linha da cintura, como por exemplo: placas, orelhões, arvores, entre outros.

Critérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um obstáculo a sua frente e desviar do orelhão.

**Figura 20.** Cenário de teste para sensor da cabeça



[Fonte: Elaboração própria]



## 5.3 Cenário C

O Cenário C apresentado pela Figura 21, tem por objetivo, representar situações onde o deficiente visual possa vir a colidir com obstáculos abaixo da linha da cintura, como por exemplo: bancos, batentes, entre outro.

Critérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um obstáculo a sua frente e desviar do batente.

**Figura 21.** Cenário de teste para sensores das pernas



[Fonte: Elaboração própria]

## 5.4 Cenário D

O Cenário D apresentado pela Figura 22, tem por objetivo, representar situações onde o deficiente visual possa vir a encontrar em seu trajeto uma superfície que possua um pequeno desnível.

Critérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um desnível a sua frente e conseguir atravessá-lo, sem tropeçar ou cair.

**Figura 22.** Primeiro cenário de teste para sensores detectores de desniveis



[Fonte: Elaboração própria]

## 5.5 Cenário E

O Cenário E apresentado pela Figura 23, tem por objetivo, representar situações onde o deficiente visual possa vir a encontrar em seu trajeto batentes. O obstáculo utilizado no teste possui 15cm de altura.

Crítérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um batente a sua frente e consiga atravessá-lo sem tropeçar.

**Figura 23.** Segundo cenário de teste para sensores detectores de desniveis



[Fonte: Elaboração própria]

## 5.6 Cenário F

O Cenário F apresentado pela Figura 24, tem por objetivo, representar situações onde o deficiente visual possa vir a encontrar em seu trajeto uma superfície que possua um grande desnível.

Critérios de aceitação: o portador do dispositivo deve ser capaz de identificar que existe um desnível a sua frente e conseguir atravessá-lo, sem tropeçar ou cair.

**Figura 24.** Terceiro cenário de teste para sensores detectores de desniveis



[Fonte: Elaboração própria]

# Capítulo 6

## Resultados obtidos

Para efetuar o teste do dispositivo, foram submetidos três voluntários, dois deles vendados e equipados com o dispositivo, identificados como Pessoa A e Pessoa B, e um voluntário sem estar vendado e sem o dispositivo, identificado como Pessoa C, aos cenários descritos no capítulo 5.

Os resultados obtidos referentes ao tempo para cruzar o obstáculo, e o sucesso na atividade estão descritos nas Tabelas 3 a 5:

**Tabela 3.** Resultados Pessoa A

Cenário	Tempo(s)	Cumpriu o objetivo	Observações
A	18.7	Sim	Demonstrou facilidade em não colidir com o obstáculo
B	5.3	Sim	Demonstrou um pouco de dificuldade para conseguir desviar do obstáculo
C	5.9	Sim	Demonstrou facilidade em desviar do obstáculo
D	3.9	Sim	Demonstrou facilidade em não tropeçar com o desnível
E	-	Não	Tropeçou no batente
F	-	Não	Tropeçou ao sentir o desnível

[Fonte: Autoria Própria]

**Tabela 4.** Resultados Pessoa B

Cenário	Tempo(s)	Cumpriu o objetivo	Observações
A	17.8	Sim	Demonstrou facilidade em não colidir com o obstáculo
B	4.3	Sim	Demonstrou facilidade em desviar do obstáculo
C	4.5	Sim	Demonstrou facilidade em desviar do obstáculo
D	3.7	Sim	Demonstrou facilidade em não tropeçar com o desnível
E	6.0	Sim	Demonstrou muita dificuldade para cruzar o batente
F	-	Não	Tropeçou ao sentir o desnível

[Fonte: Autoria Própria]

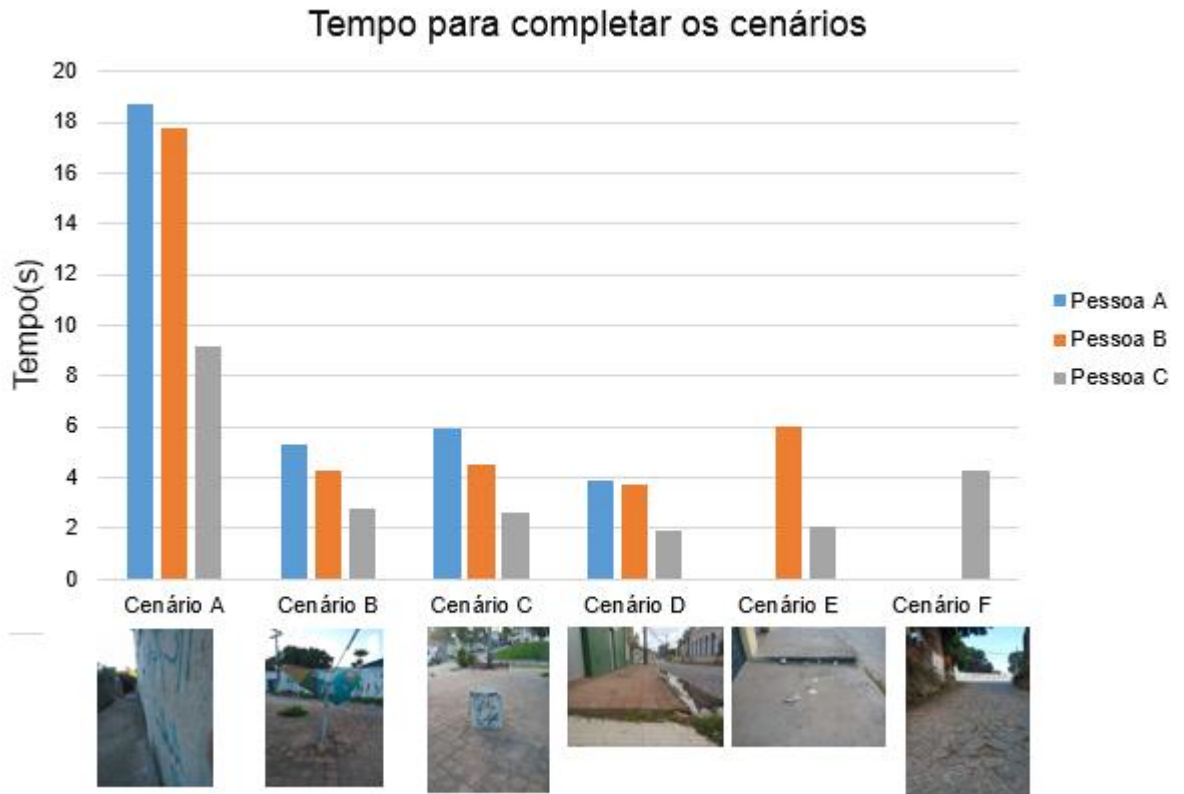
**Tabela 5.** Resultados Pessoa C

Cenário	Tempo(s)
A	9.2
B	2.8
C	2.6
D	1.9
E	2.1
F	4.3

[Fonte: Autoria Própria]

A partir dos resultados das Tabelas 3 a 5, foi elaborado o gráfico presente na Figura 25 com o objetivo de comparar os dados obtidos.

**Figura 25.** Gráfico comparativo dos testes efetuados



[Fonte: Autoria Própria]

O dispositivo proposto mostrou-se bastante eficiente para identificação e notificação dos obstáculos presentes nos cenários A a D para as pessoas A e B, isto é, obteve um bom aproveitamento para identificar obstáculos laterais, obstáculos abaixo da linha da cintura e pequenos desníveis redor, em um tempo cerca de 50% maior do que uma pessoa sem venda.

Entretanto, para os cenários E e F, o sistema não foi capaz de auxiliar os portadores do dispositivo a concluírem os objetivos propostos no teste.

# Capítulo 7

## Considerações finais

### 7.1 Conclusão

Diante do cenário de falta de acessibilidade nas ruas e grande número de deficientes visuais, esse trabalho propôs um sistema embarcado inteligente e vestível, com o objetivo de auxiliar a locomoção de deficientes visuais, notificando-os sobre possíveis obstáculos ao seu redor.

Ao comparar os resultados obtidos a partir dos testes feitos nos 6 cenários, com a finalidade de representar situações a qual o deficiente visual pode se deparar em seu cotidiano, conclui-se que o protótipo construído obteve resultados satisfatórios nos cenários A à D, porém, para os cenários E e F, necessita-se de mais estudos, de modo a verificar se variações de parâmetros podem melhorar os resultados obtidos.

Em relação ao controle utilizando redes neurais, foi possível verificar que o mesmo não trouxe resultados significativos para diminuição do tempo entre duas medições de um sensor que captou um obstáculo próximo, porém, permitiu que o script utilizado no arduino ficasse mais simples, além de tornar mais fácil mudanças nas regras utilizadas, tornando-se necessária apenas a mudança no conjunto de exemplos apresentados a rede neural.

Por fim, conclui-se também que é possível a criação de um dispositivo vestível, controlado por uma rede neural MLP, que capte obstáculos e desníveis próximos ao deficientes visuais, utilizando sensores ultrassônicos dispostos em várias partes do corpo, e que consiga notificar o portador do dispositivo, através de atuadores vibratórios acionados com sinais PWM, quão próximo estão esses possíveis obstáculos a seu redor.



## 7.2 Trabalhos futuros

Para trabalhos futuros, pretende-se miniaturizar o protótipo construído, além de utilizar uma conexão *Wireless*, seja por *Bluetooth* ou radio-frequência, para comunicação entre os sensores e a central de processamento, tornando o dispositivo mais confortável para o usuário, e eliminando também, as situações onde os fios possam enganchar em algum obstáculo.

Almeja-se, também, realizar experimentos que envolvam o monitoramento de deficientes visuais em ambientes reais durante as suas atividades cotidianas, de modo a validar o dispositivo proposto.

Por fim, pretende-se também viabilizar a integração do dispositivo com *smartphones*, de modo a dividir o processamento e possibilitar integrações com sistemas como:

- Sistema para identificação de linhas de ônibus;
- Sistema para identificação do estado do sinal de trânsito;
- Sistema para geolocalização do portador do dispositivo pela família;

# Bibliografia

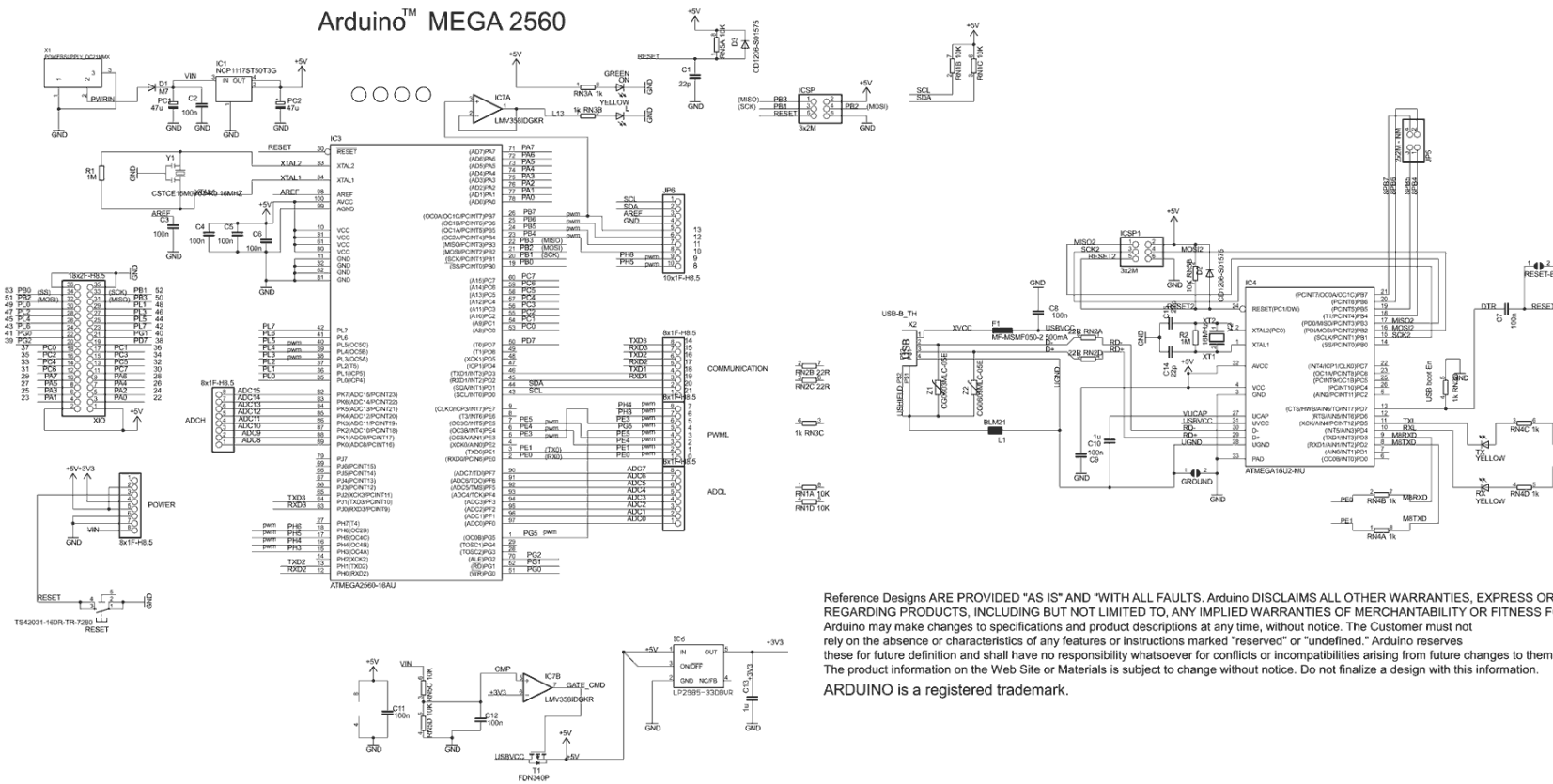
- [1] AnnuItWalk. Disponível em: < <http://annuitwalk.com/> > Acessado em: 12 de Junho de 2016.
- [2] Arduino Mega. Pagina principal do Arduino Mega. Disponível em: <<https://www.arduino.cc/en/Main/ArduinoBoardMega>> Acessado em: 06 de Junho de 2016
- [3] Arduino. Pagina principal do Arduino. Disponível em: <<https://www.arduino.cc/>> Acessado em: 06 de Junho de 2016
- [4] BALL, Stuart. – “Embedded Microprocessor Systems: Real World Design”, 3rd edition, Editora: MCPros, EUA, 2005
- [5] BERSCH, Rita: Introdução a Tecnologia Assistiva – CEDI – Centro Especializado em Desenvolvimento Infantil, Porto Alegre, 2008.
- [6] Brasil Telemedicina. Acuidade Visual. Disponível em: <<http://brasiltelemedicina.com.br/nossos-produtos/acuidade-visual>> Acesso em 05 de Junho de 2016
- [7] BRASIL. SDHPR - Secretaria Nacional de Promoção dos Direitos da Pessoa com Deficiência - SNPDP. 2009. Disponível em: <<http://www.pessoacomdeficiencia.gov.br/app/publicacoes/tecnologia-assistiva>> Acesso em 05 de Junho de 2016.
- [8] CARDOZO BUENO, Alessandro. Bengala Eletrônica para Deficientes Visuais. 2010. Trabalho de Conclusão de Curso de Engenharia da Computação, Universidade Positivo, Curitiba.
- [9] DINIZ, E. H; MEDEIROS DEMARCHI, Felipe. Desenvolvimento de um sistema para auxílio à locomoção de deficientes visuais através da implementação em arquiteturas reconfiguráveis da transformada Census para estimação de distância usando visão estéreo. 2013. Trabalho de Conclusão de Curso em Engenharia Eletrônica. Universidade de Brasília, Brasília.

- [10] DR VISAIO. Doenças oftamologicas. Disponível em: <<http://www.drvisao.com.br/conheca/Doencas-Oftalmologicas/44-Cegueira>> Acesso em 05 de Maio de 2016.
- [11] GONÇALVES, E. A. Sensor Ultrassônico para Deficientes Visuais. 2010.
- [12] GUEDE, José Ricardo Abalde; MADRUGA, Cleber Luis Mendes; Silva, Tatiana dos Reis. MEDIDOR DE NÍVEL ULTRA-SONICO. XIII INIC, IX EPG E III INIC JR - UNIVAP, São José dos Campos, São Paulo, 2009
- [13] HALL, Eldon C. Journey to the Moon: The History of the Apollo Guidance Computer .(Reston, Virginia, USA: American Institute of Aeronautics and Astronautics). 1996. pp. 196
- [14] HC-SR04. Robocore - Página de venda do sensor. Disponível em: <[https://www.robocore.net/modules.php?name=GR\\_LojaVirtual&prod=620](https://www.robocore.net/modules.php?name=GR_LojaVirtual&prod=620)> Acessado em: 06 de Junho de 2016
- [15] Instituto Benjamin Constant. Pessoa Portadora de deficiência. Disponível em: <<http://www.ibc.gov.br/?itemid=396>> Acesso em 30 de Abril de 2016.
- [16] Instituto Benjamin Constant. Reações Psicológicas à Perda da Visão. Disponível em: <<http://www.ibc.gov.br/?itemid=110>> Acesso em 20 de Abril de 2016.
- [17] Instituto Brasileiro de Geografia e Estatística. Censo 2010. Disponível em: <<http://www.ibge.gov.br/apps/snig/v1/?loc=0&cat=-1,-2,-3,128&ind=4642>> Acesso em: 25 de Abril de 2016.
- [18] MCCULLOCH, W.S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. Bulletin os Mathematical Biophisics, 1943, p.115-133.
- [19] NAVARRO, L. F. "Redes Neurais Artificiais Aplicadas a Detecção, Classificação e Localização de Faltas em Sistemas de Distribuição de Energia Elétrica". Trabalho de Conclusão de Curso - Escola de Engenharia da São Carlos, Universidade de São Paulo, 2014.

- [20] Organização das Nações Unidas. OMS afirma que existem 39 milhões de cegos no mundo. Disponível em: <<https://nacoesunidas.org/oms-afirma-que-existem-39-milhoes-de-cegos-no-mundo/>> Acesso em: 23 de Abril de 2016.
- [21] PUPO, Deise Tallarico; MELO, Amanda Meincke; PÉREZ FERRÉS, Sofia. Acessibilidade: discurso e prática no cotidiano das bibliotecas . São Paulo, SP: UNICAMP, 2008. pp 94.
- [22] PWM. Conceitos e Circuito Exemplo. Disponível em: <<https://www.eecis.udel.edu/~portnoi/academic/academic-files/pwm.html>> Acessado em: 06 de Junho de 2016
- [23] RADABAUGH, M. P. NIDRR's Long Range Plan - *Technology for Access and Function Research*. Seção 2: *NIDRR Research Agenda* Capítulo 5: *TECHNOLOGY FOR ACCESS AND FUNCTION*
- [24] RIBEIRO FILHO, J. S. ; ARAUJO, R. B. ; CAVALCANTE, R. V. ; BAGANHA, A. N. A. B. . ARGOS Auxílio à Locomoção de Deficientes Visuais a partir de Pulseira Microcontrolada. In: *Computer On The Beach*, 2011, Florianópolis. ANAIS DE 2011.
- [25] SCHULZ SEFRIN, Felipe. SACCOB: Sistema de Alerta Contra Colisão com Obstáculos. 2012. Trabalho de Conclusão de Curso de Bacharelado em Ciência da Computação, Universidade Estadual do Oeste do Paraná.
- [26] SILVA, I. N. da; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas: curso prático. São Paulo, SP: Artliber, 2010
- [27] Sociedade Brasileira de Visão Subnormal. Classes de acuidade visual. Disponível em: <<http://www.cbo.com.br/subnorma/conceito.htm>> Acesso em 10 de Maio de 2016.
- [28] TANENBAUM, A. S. Sistemas operacionais modernos. 3.ed. São Paulo: Livro Técnico, Prentice Hall, 2003
- [29] VALENÇA, M. Fundamentos de Redes Neurais:Livro Rápido,2ª Ed, 2014.

# Anexo A

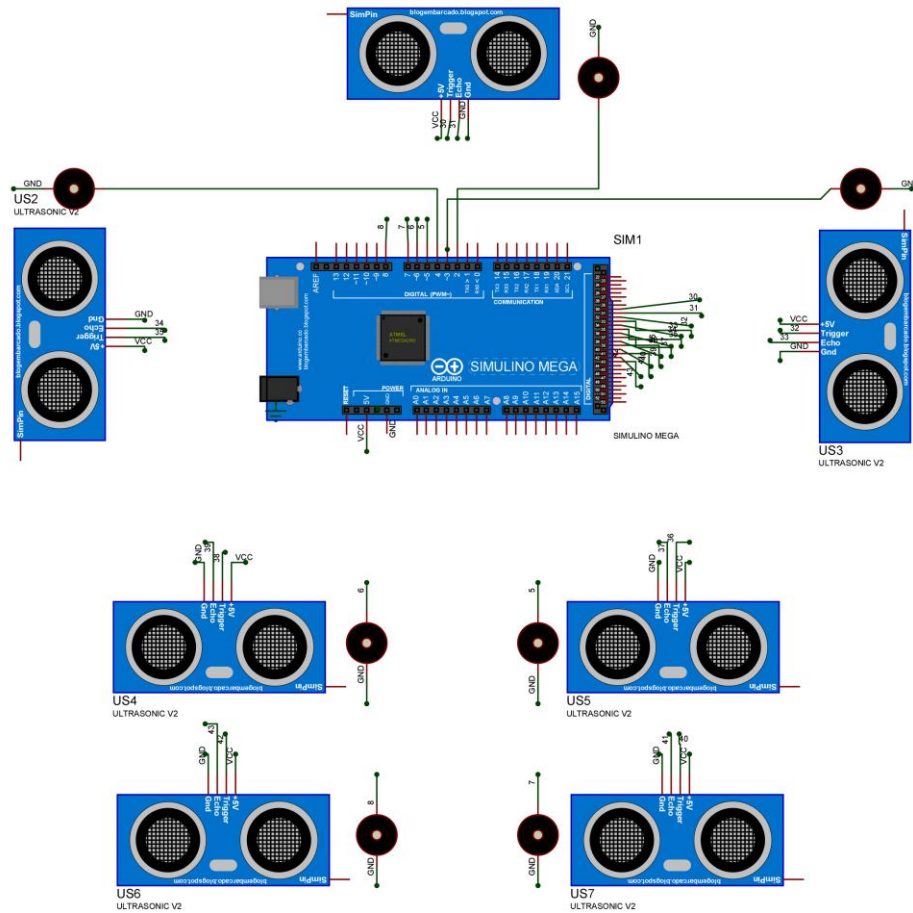
## Datasheet Arduino Mega



Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information. ARDUINO is a registered trademark.

# Apêndice B

## Diagrama elétrico



# Apêndice C

## Código Fonte da Rede Neural

```
#include <math.h>

#define MAXCICLO 100000
#define NUMEXEMPLOS 100
#define QTDENTRADAS 8
#define QTDNEURONIOSESCONDIDOS 17
#define QTDSAIDAS 8
#define ALFA 0.5

float pesosCamada1[QTDENTRADAS][QTDNEURONIOSESCONDIDOS+1];
float pesosCamada2[QTDNEURONIOSESCONDIDOS+1][QTDSAIDAS];
float Escondida[QTDNEURONIOSESCONDIDOS+1];
float Saidas[QTDSAIDAS];
float erroIntermediario[QTDNEURONIOSESCONDIDOS+1];
float erroSaida[QTDSAIDAS];
float sensibilidadeIntermediario[QTDNEURONIOSESCONDIDOS+1];
float sensibilidadeSaida[QTDSAIDAS];

byte vetorIndiceExemplo[NUMEXEMPLOS];

int auxIndiceExemplo;
int i, j, e;
float net, erroMedioQuadratico;
int cicloTreinamento;

byte Entradas[NUMEXEMPLOS][QTDENTRADAS] = { ... };

byte SaidaDesejada[NUMEXEMPLOS][QTDSAIDAS] = { ... };

void setup(){
  Serial.begin(9600);
  randomSeed(millis());
  InicializarRede();
  InicializarVetorExemplo();
}

void InicializarVetorExemplo(){
  for (i=0; i< NUMEXEMPLOS; i++){
    vetorIndiceExemplo[i]=1;
  }
}

void InicializarRede(){
  for( i = 0 ; i <QTDENTRADAS ; i++ ) {
    for( j = 0 ; j < QTDNEURONIOSESCONDIDOS +1 ; j++ ) {
```

```

    pesosCamada1[i][j] = (float) random(-0.5, 0.5);
}
}

for( i = 0 ; i < QTDNEURONIOSESCONDIDOS+1; i ++ ) {
    for( j = 0 ; j < QTDSAIDAS; j++ ) {
        pesosCamada2[i][j] = (float) random(-0.5, 0.5);
    }
}

Escondida[0]=1; //Bias
}

int sortearPosicao(){
    int s = random(0, NUMEXEMPLOS);

    while(vetorIndiceExemplo[s]!=1){
        s--;
        if(s<0){
            s = NUMEXEMPLOS-1;
        }
    }

    vetorIndiceExemplo[s]=0;
    auxIndiceExemplo--;
    return s;
}

void calcularExemplo(int e){

    //imprimirEntrada(e);
    //imprimirSaidaDesejada(e);

    //calcula sinais da camada Escondida para exemplo e
    for( i = 1 ; i < QTDNEURONIOSESCONDIDOS +1; i++ ) {
        net = pesosCamada1[0][i];
        for( j = 1 ; j < QTDENTRADAS ; j++ ) {
            net += Entradas[e][j] * pesosCamada1[j][i] ;
        }
        Escondida[i] = 1.0/(1.0 + exp(-net)) ;
    }

    //calcula sinais da camada de saida para exemplo e
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        net = pesosCamada2[0][i] ;
        for( j = 1 ; j < QTDNEURONIOSESCONDIDOS +1; j++ ) {
            net += Escondida[j] * pesosCamada2[j][i] ;
        }
        Saidas[i] = round(1.0/(1.0 + exp(-net)));
        //Saidas[i] = 1.0/(1.0 + exp(-net));
    }

    //imprimirSaidaCalculada();
}

```



```

}

void calcularErroSaida(int e){
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        erroSaida[i] = (SaidaDesejada[e][i] - Saidas[i]);
    }
}

void calcularErroMedio(){
    erroMedioQuadratico = 0.0;
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        erroMedioQuadratico += (Saidas[i]*Saidas[i]);
    }
    erroMedioQuadratico = erroMedioQuadratico/QTDSAIDAS;
}

void calcularSensibilidadeSaida(){
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        sensibilidadeSaida[i] = Saidas[i]*(1-Saidas[i])*erroSaida[i];
    }
}

void calcularSensibilidadeEscondida(){
    float aux;
    for( j = 0 ; j < QTDNEURONIOSESCONDIDOS +1; j++ ) {
        aux = 0.0;
        for( i = 0 ; i < QTDSAIDAS ; i++ ) {
            aux += (sensibilidadeSaida[i]*pesosCamada2[j][i]);
        }
        sensibilidadeIntermediario[j] = Escondida[j]*(1-Escondida[j])*aux;
    }
}

void ajustePesosCamada2(){
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        for( j = 0 ; j < QTDNEURONIOSESCONDIDOS +1; j++ ) {
            pesosCamada2[j][i] = pesosCamada2[j][i] +
            (ALFA*sensibilidadeSaida[i]*Escondida[j]*(1-Escondida[j]) );
        }
    }
}

void ajustePesosCamada1(int e){
    for( i = 0 ; i < QTDETRADAS ; i++ ) {
        for( j = 0 ; j < QTDNEURONIOSESCONDIDOS +1; j++ ) {
            pesosCamada1[i][j] = pesosCamada1[i][j] +
            (ALFA*sensibilidadeIntermediario[j]*Entradas[e][i]);
        }
    }
}

void imprimirEntrada(int e){
    Serial.print ("Entradas: ");
    for( i = 0 ; i < QTDETRADAS ; i++ ) {

```

```
        Serial.print (Entradas[e][i]);
        Serial.print (" ");
    }
    Serial.println("");
}

void imprimirSaidaDesejada(int e){
    Serial.print ("Saida Desejada: ");
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        Serial.print (SaidaDesejada[e][i]);
        Serial.print (" ");
    }
    Serial.println("");
}

void imprimirSaidaCalculada(){
    Serial.print ("Saidas calculadas:");
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        Serial.print (Saidas[i]);
        Serial.print (" ");
    }
    Serial.println("");
}

void imprimirPesoCamada1(){
    Serial.println("Camada 1");
    for( i = 0 ; i < QTDESTRADAS ; i ++ ) {
        for( j = 0 ; j < QTDNEURONIOSESCONDIDOS +1; j++ ) {
            Serial.print("W");
            Serial.print(i);
            Serial.print(j);
            Serial.print(":");
            Serial.print(pesosCamada1[i][j]);
            Serial.println("");
        }
    }
}

void imprimirPesoCamada2(){
    Serial.println("Camada 2");
    for( i = 0 ; i < QTDNEURONIOSESCONDIDOS +1; i ++ ) {
        for( j = 0 ; j < QTDSAIDAS; j++ ) {
            Serial.print("W");
            Serial.print(i);
            Serial.print(j);
            Serial.print(":");
            Serial.print(pesosCamada2[i][j]);
            Serial.println("");
        }
    }
}

void loop (){
```

```
for( cicloTreinamento = 0 ; cicloTreinamento < MAXCICLO ; cicloTreinamento++) {  
    auxIndiceExemplo = NUMEXEMPLOS;  
    while(auxIndiceExemplo !=0) {  
        e = sortearPosicao();  
        calcularExemplo(e);  
        calcularErroSaida(e);  
        calcularSensibilidadeSaida();  
        calcularSensibilidadeEscondida();  
        ajustePesosCamada2();  
        ajustePesosCamada1(e);  
    }  
    InicializarVetorExemplo();  
}  
  
imprimirPesoCamada1();  
imprimirPesoCamada2();  
delay(1000);  
}
```

# Apêndice D

## Exemplo dados utilizados para treinamento

Entradas								Saídas							
Bias (X0)	Cabeça (X1)	BraçoEsq (X2)	BraçoDir (X3)	PernaEsq (X4)	PernaDir (X5)	PeEsq (X6)	PeDir (X7)	Coleta1 (Y0)	Coleta2 (Y1)	Coleta3 (Y2)	Coleta4 (Y3)	Coleta5 (Y4)	Coleta6 (Y5)	Coleta7 (Y6)	Coleta8 (Y7)
1	32	50	60	100	79	50	50	1	2	3	1	4	5	6	7
1	31	60	50	100	73	45	50	1	3	2	1	4	5	6	7
1	40	80	80	200	129	25	26	1	5	6	1	2	3	4	7
1	20	120	80	30	104	50	45	1	4	5	1	6	3	2	7
1	50	120	120	120	147	70	80	1	6	5	1	2	3	4	7
1	121	12	63	54	57	160	76	6	2	7	6	4	5	3	1
1	14	49	124	124	191	89	198	7	6	1	7	2	3	4	5
1	35	101	133	129	45	136	56	6	1	3	6	2	4	5	7
1	173	174	190	109	129	166	135	6	7	4	6	1	2	3	5
1	40	124	31	83	169	156	185	7	6	3	7	1	2	4	5

# Apêndice E

## Código Fonte do Arduino

```
#include <Ultrasonic.h>
#include <avr/wdt.h>
#include <math.h>

#define cabeca 2
#define braEsq 3
#define braDir 4
#define perEsq 5
#define perDir 6
#define peEsq 7
#define peDir 8
#define minDistancia 180
#define maxDistancia 2
#define minDeclividade 60
#define maxDeclividade 120
#define minBatente 20
#define maxBatente 5
#define QTDENTRADAS 8
#define QTDNEURONIOSESCONDIDOS 15
#define QTDSAIDAS 8

Ultrasonic ultracabeca(30,31); // (Trig PIN,Echo PIN)
Ultrasonic ultraBraEsq(32,33); // (Trig PIN,Echo PIN)
Ultrasonic ultraBraDir(34,35); // (Trig PIN,Echo PIN)
Ultrasonic ultraPerEsq(36,37); // (Trig PIN,Echo PIN)
Ultrasonic ultraPerDir(38,39); // (Trig PIN,Echo PIN)
Ultrasonic ultraPeEsq(40,41); // (Trig PIN,Echo PIN)
Ultrasonic ultraPeDir(42,43); // (Trig PIN,Echo PIN)

int med=0;
int mapeado=0;
long microsec=0;
int i, j;

float Intermediaria[QTDNEURONIOSESCONDIDOS+1];
byte Saidas[QTDSAIDAS] = {1,2,3,1,4,5,6,7};
float pesosCamada1[QTDENTRADAS][QTDNEURONIOSESCONDIDOS+1] ={}; //pesos
otimos obtidos
float pesosCamada2[QTDNEURONIOSESCONDIDOS+1][QTDSAIDAS] ={}; //pesos otimos
obtidos
byte Entradas[QTDENTRADAS] = {1,1,1,1,1,1,1,1};

void setup() {
  Serial.begin(9600);
  pinMode(cabeca, OUTPUT); //Cabeça
  pinMode(braEsq, OUTPUT); //Braço Esquerdo
```

```

pinMode(braDir, OUTPUT); //Braço Direito
pinMode(perEsq, OUTPUT); //Perna Esquerda
pinMode(perDir, OUTPUT); //Perna Direita
pinMode(peEsq, OUTPUT); //Pé Esquerdo
pinMode(peDir, OUTPUT); //Pe Direito

analogWrite(cabeça, 0); //Cabeça
analogWrite(braEsq, 0); //Braço Esquerdo
analogWrite(braDir, 0); //Braço Direito
analogWrite(perEsq, 0); //Perna Esquerda
analogWrite(perDir, 0); //Perna Direita
analogWrite(peEsq, 0); //Pé Esquerdo
analogWrite(peDir, 0); //Pe Direito
wdt_enable(WDTO_4S); // habilita o watchdog
}

void calcularExemplo(){
    float net;
    //calcula sinais da camada Intermediaria para exemplo e
    for( i = 1 ; i < QTNEURONIOSESCONDIDOS +1; i++ ) {
        net = pesosCamada1[0][i];
        for( j = 1 ; j < QTDENTRADAS ; j++ ) {
            net += Entradas[j] * pesosCamada1[j][i] ;
        }
        Intermediaria[i] = 1.0/(1.0 + exp(-net)) ;
    }

    //calcula sinais da camada de saida para exemplo e
    for( i = 0 ; i < QTDSAIDAS ; i++ ) {
        net = pesosCamada2[0][i] ;
        for( j = 1 ; j < QTNEURONIOSESCONDIDOS +1; j++ ) {
            net += Intermediaria[j] * pesosCamada2[j][i] ;
        }
        Saidas[i] = round(1.0/(1.0 + exp(-net)));
    }
}

void loop(){

    for( i=0; i<QTDSAIDAS; i++){

        if(Saidas[i]==7){
            med=0;
            microsec = ultraPeEsq.timing();
            med=ultraPeEsq.convert(microsec, Ultrasonic::CM); // CM or INC
            delay(2);
            microsec = ultraPeEsq.timing();
            med=med+ultraPeEsq.convert(microsec, Ultrasonic::CM); // CM or INC
            delay(2);
            microsec = ultraPeEsq.timing();
            med=med+ultraPeEsq.convert(microsec, Ultrasonic::CM); // CM or IN
            med = med/3;
            Entradas[7] = med;
            if(med<=minBatente && med>=maxBatente){

```

```

        mapeado= map(med,maxBatente ,minBatente ,0,255);
    }else if(med<=maxDeclividade && med>=minDeclividade){
        mapeado= map(med,minDeclividade ,maxDeclividade ,0,255);
    }else{
        mapeado=0;
    }
    analogWrite(peEsq, mapeado); //Pé Esquerdo
    Serial.print("Pe Esq:");
    Serial.println(med);
    delay(50);

}else if(Saidas[i]==2){
    med=0;
    microsec = ultraBraEsq.timing();
    med=ultraBraEsq.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraBraEsq.timing();
    med=med+ultraBraEsq.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraBraEsq.timing();
    med=med+ultraBraEsq.convert(microsec, Ultrasonic::CM); // CM or INC
    med = med/3;
    Entradas[2] = med;
    mapeado= map(med,maxDistancia ,minDistancia ,0,255);
    analogWrite(braEsq, mapeado); //Braço Esquerdo
    Serial.print("Braco Esq:");
    Serial.println(med);
    delay(50);

}else if(Saidas[i]==3){

    med=0;
    microsec = ultraBraDir.timing();
    med=ultraBraDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraBraDir.timing();
    med=med+ultraBraDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraBraDir.timing();
    med=med+ultraBraDir.convert(microsec, Ultrasonic::CM); // CM or INC
    med = med/3;
    Entradas[3] = med;
    mapeado= map(med,maxDistancia ,minDistancia ,0,255);
    analogWrite(braDir, mapeado); //Braço Direito
    Serial.print("Braco Dir:");
    Serial.println(med);
    delay(50);

}else if(Saidas[i]==4){

    med=0;
    microsec = ultraPerEsq.timing();
    med=ultraPerEsq.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);

```

```

    microsec = ultraPerEsq.timing();
    med=med+ultraPerEsq.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraPerEsq.timing();
    med=med+ultraPerEsq.convert(microsec, Ultrasonic::CM); // CM or IN
    med = med/3;
    Entradas[4] = med;
    mapeado= map(med,maxDistancia ,minDistancia ,0,255);
    analogWrite(perEsq, mapeado); //Perna Esquerda
    Serial.print("Perna Esq:");
    Serial.println(med);
    delay(50);

}

}else if(Saidas[i]==5){

    med=0;
    microsec = ultraPerDir.timing();
    med=ultraPerDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraPerDir.timing();
    med=med+ultraPerDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraPerDir.timing();
    med=med+ultraPerDir.convert(microsec, Ultrasonic::CM); // CM or IN
    med = med/3;
    Entradas[5] = med;
    mapeado= map(med,maxDistancia ,minDistancia ,0,255);
    analogWrite(perDir, mapeado); //Perna Direita
    Serial.print("Perna Dir:");
    Serial.println(med);
    delay(50);

}

}else if(Saidas[i]==6){

    med=0;
    microsec = ultraPeDir.timing();
    med=ultraPeDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraPeDir.timing();
    med=med+ultraPeDir.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultraPeDir.timing();
    med=med+ultraPeDir.convert(microsec, Ultrasonic::CM); // CM or IN
    med = med/3;
    Entradas[6] = med;
    if(med<=minBatente && med>=maxBatente){
        mapeado= map(med,maxBatente ,minBatente ,0,255);
    }else if(med<=maxDeclividade && med>=minDeclividade){
        mapeado= map(med,minDeclividade ,maxDeclividade ,0,255);
    }else{
        mapeado=0;
    }
    analogWrite(peDir, mapeado); //Pe Direito
    delay(50);
}

```



```
Serial.print("Pe Dir:");
Serial.println(med);

}else{

    med=0;
    microsec = ultracabeca.timing();
    med=ultracabeca.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultracabeca.timing();
    med=med+ultracabeca.convert(microsec, Ultrasonic::CM); // CM or INC
delay(2);
    microsec = ultracabeca.timing();
    med=med+ultracabeca.convert(microsec, Ultrasonic::CM); // CM or INC
    med = med/3;
    Entradas[1] = med;
    mapeado= map(med,maxDistancia ,minDistancia ,0,255);
    analogWrite(cabeca, mapeado); //Cabeça
    Serial.print("Cabeça:");
    Serial.println(med);
    delay(50);
}
}
wdt_reset(); // reseta o watchdog, ou seja se o programa travar e não ativar esse reset
em 4
calcularExemplo();
}
```