

**CENTRO PAULA SOUZA**  
**FACULDADE DE TECNOLOGIA DE SANTO ANDRÉ**  
**Tecnologia em Mecatrônica Industrial**

**Tamires dos Santos**

**DISPOSITIVO DE INTERPRETAÇÃO DE ONDAS CEREBRAIS PARA  
CONTROLE DE UM BRAÇO ROBÓTICO**

**Santo André - SP**

**2018**

**Tamires dos Santos**

**DISPOSITIVO DE INTERPRETAÇÃO DE ONDAS CEREBRAIS PARA  
CONTROLE DE UM BRAÇO ROBÓTICO**

Monografia apresentada ao Curso de Tecnologia em Mecatrônica Industrial da FATEC Santo André como requisito parcial para obtenção do título de Tecnólogo em Mecatrônica Industrial.

Orientador: Prof. Me. Paulo Tetsuo Hoashi.

Coorientador: Prof. Me. Murilo Zanini de Carvalho.

**Santo André - SP**

**2018**

## FICHA CATALOGRÁFICA

S237d

Santos, Tamires dos

Dispositivo de interpretação de ondas cerebrais para controle de um braço robótico / Tamires dos Santos. - Santo André, 2018. – 86f: il.

Trabalho de Conclusão de Curso – FATEC Santo André.  
Curso de Tecnologia em Mecatrônica Industrial, 2018.

Orientador: Prof. Me. Paulo Tetsuo Hoashi

1. Mecatrônica. 2. Cérebro humano. 3. Braço robótico. 4. Pulsos elétricos. 5. Dispositivo Mindflex. 6. Capacete. 7. Atuadores. 8. Processamento. I. Dispositivo de interpretação de ondas cerebrais para controle de um braço robótico.

629.8

## LISTA DE PRESENÇA

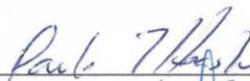
SANTO ANDRÉ, 27 DE JUNHO DE 2018

LISTA DE PRESENÇA REFERENTE À APRESENTAÇÃO DO  
TRABALHO DE CONCLUSÃO DE CURSO COM O TEMA  
“DISPOSITIVO DE INTERPRETAÇÃO DE ONDAS CEREBRAIS  
PARA CONTROLE DE UM BRAÇO ROBÓTICO” DA ALUNA DO  
6º SEMESTRE DESTA U.E.

**BANCA**

PRESIDENTE:

PROF. PAULO TETSUO HOASHI



MEMBROS:

PROF. FERNANDO GARUP DALBO



PROF. ROGÉRIO DANTAS

**ALUNA:**

TAMIRES DOS SANTOS



## DEDICATÓRIA

Dedico este trabalho aos meus familiares e a todas as pessoas que me ajudaram no desenvolvimento deste projeto.

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por me conceder a oportunidade de realizar o desenvolvimento deste projeto e por conseguir entrar nesta instituição maravilhosa.

Agradeço a minha família que me incentivaram em todas as minhas decisões, me ajudando e sempre estando ao meu lado.

Agradeço aos meus orientadores, Paulo e Murilo, por estarem ao meu lado no desenvolvimento deste projeto, por terem paciência comigo e por me darem algo mais preciso que tenho em minha vida, o conhecimento.

Agradeço a todos os professores que fizeram parte desta etapa da minha vida: Priscilla, Regiane, Celso Tabajara, Edson, Eliel, Fernando Garup, Francisco, Luiz Vasco, Moacyr, Nelson Lavecchia, Pedro Galani, Roberto Bortolussi, Rogério, Valter Thomaz, Wellington.

Agradeço a todos que fazem parte da Fatec Santo André.

E agradeço ao Clodoaldo Dorini Guerios por emprestar a prótese para o início dos testes, muito obrigada.

## RESUMO

O projeto “Dispositivo de Interpretação de Ondas Cerebrais para Controle de um Braço Robótico” tem como objetivo coletar os pulsos elétricos emitidos pelo cérebro humano, processar estes sinais e transforma-los em uma ação no braço robótico (prótese). As coletas destes sinais foram realizadas utilizando o capacete do dispositivo *MindFlex* (da empresa Neurosky), conectado a um Raspberry Pi 3. A rede neural artificial, RNA possui a capacidade de interpretar o movimento que o indivíduo (dentro dos movimentos já predefinidos no período de testes), foi implementada na Raspberry. A rede consegue classificar o sinal recebido utilizando um algoritmo de classificação, que foi treinado pelo conjunto de dados, *dataset*, coletado pelo capacete. Na saída desta rede neural está a prótese, que recebeu como sinal de ativação, os dados de saída da rede neural. Os sinais emitidos pelo indivíduo, na forma de pensamento, foram reproduzidos pela prótese considerando os movimentos previamente treinados. Assim, podendo ser aplicado ao qualquer indivíduo desde que haja um período de coleta e análise dos dados antes de iniciar a implementação do projeto em outros indivíduos.

Palavras Chave: Cérebro. RNA. Processamento. Atuador. Sinais. Classificação.

## **ABSTRACT**

The project "Brain Wave Interpretation Instrument to Control a Robotic Arm" aims to collect the brain impulses from the human brain, process these signals and transform the movements into an action on the robotic arm (prosthesis). The collections were captured using the MindFlex device (from the company Neurosky), connected to a Raspberry Pi 3. An artificial neural network, the RNA has an ability to interpret the individual's movement. testis), has been implemented in Raspberry. The data exchange through a classification algorithm was trained by the data set, set of data, collected by the helmet. At the output of this neural network is a prosthesis, which received as activation signal, the output data of the neural network. The signs emitted by the individual, in the form of thought, were reproduced by the prosthesis in relation to the trained ones. Thus, they will be applied at any other time as long as there is a collection and analysis period of the previous data from one project implementation to another.

Keywords: Brain. RNA. Processing. Actuator. Signals. Ranking.

## LISTA DE ILUSTRAÇÕES

Figura 2. 1 - Sistema Interfaces Cérebro Computador da FCEE.....	19
Figura 2. 2 - Robô para ser “vestido” por pessoas paraplégicas. ....	20
Figura 2. 3 - Teclado adaptado. ....	22
Figura 2. 4 - Lobos Cerebrais.....	24
Figura 2. 5 - Estrutura de um neurônio.....	25
Figura 2. 6 - Sinais registrados por Hans Berger.....	25
Figura 2. 7 - Onda Gamma. ....	27
Figura 2. 8 - Onda Beta.....	27
Figura 2. 9 - Onda Alpha.....	28
Figura 2. 10 - Onda Theta.....	28
Figura 2. 11 - Onda Delta.....	28
Figura 2. 12 - Leitor EEG da empresa EPOC Emotiv.....	30
Figura 2. 13 - Leitor EEG do dispositivo <i>MindFlex</i> .....	30
Figura 2. 14 - <i>Layout</i> do <i>hardware</i> do dispositivo <i>MindFlex</i> .....	31
Figura 2. 15 - Neurônio artificial de uma rede neural.....	33
Figura 2. 16 - Exemplo de uma RNA.....	34
Figura 2. 17 - Músculos do antebraço vista anterior. ....	36
Figura 2. 18 - Músculos do antebraço vista lateral. ....	37
Figura 2. 19 - Músculos do antebraço vista posterior. ....	39
Figura 2. 20 - Músculos da mão direita. ....	40
Figura 2. 21 - Prótese biônica. ....	41
Figura 2. 22 - Prótese mecânica. ....	42
Figura 2. 23 - Braço esquerdo do InMoov. ....	44
Figura 2. 24 - Posição dos motores na prótese. ....	45
Figura 2. 25 - Conectores do servo motor. ....	46

Figura 2. 26 - Sinais de controle do servo motor. ....	46
Figura 3. 1 – Divisão do projeto.....	47
Figura 3. 2 - Modo de coleta dos sinais com o capacete. ....	48
Figura 3. 3 - Fluxo de dados para coleta de informações.....	49
Figura 3. 4 – <i>Hardware</i> de coleta. ....	50
Figura 3. 5 – Esquema elétrico da ligação do capacete com o Arduino. ....	50
Figura 3. 6 - Estrutura da Rede Neural.....	53
Figura 3. 7 – Servo Motor.....	54
Figura 3. 8 – Divisão do projeto junto pela programação. ....	57
Figura 3. 9 – Leitura do arquivo normalizado. ....	58
Figura 3. 10 - Saída da Rede Neural.....	59
Figura 3. 11 – Realimentação do sistema. ....	60
Figura 3. 12 – Monitoramento dos programas.....	61
Figura 3. 13 – Movimento de mão aberta.....	61
Figura 3. 14 – Movimento de mão fechada. ....	62
Figura 3. 15 - Estrutura física do projeto. ....	62
Figura 4. 1 - Treinamento da rede neural.....	64
Figura 4. 2 - Classificação da rede neural. ....	64
Figura 4. 3 – Vinculo do sinal de entrada com o de saída. ....	66

## LISTA DE QUADROS

Quadro 2. 1 - Comparação dos materiais.....	43
Quadro 3. 1 – Dados coletados pelo capacete.....	49
Quadro 3. 2 - Dados com sensor <i>flex</i> .....	51
Quadro 3. 3 - Classificação dos dados para armazenamento no <i>dataset</i> .....	52
Quadro 3. 4 – Valor de cada sinal para normalizar os sinais de entrada da RNA.....	52
Quadro 3. 5 – Dados recebidos pelo módulo <i>Bluetooth</i> . ....	55
Quatro 4. 1 - Dados do <i>dataset</i> .....	63
Quatro 4. 2 - Dados do capacete. ....	65
Quatro 4. 3 - Dados normalizados.....	66

## LISTA DE ABREVIATURAS E SIGLAS

ECoG	Eletrocorticografia
EEG	Eletroencefalograma
FDM	<i>Fused Deposition Modeling</i>
GPIO	<i>General Purpose Input/Output</i>
Hz	Hertz
MLP	<i>Multi – layer Perceptron</i>
ms	Milissegundo
PC	<i>Personal Computer</i>
PWM	<i>Pulse Width Modulation</i>
RNA	Rede Neural Artificial
V	volt

# SUMÁRIO

1 INTRODUÇÃO .....	14
1.1 Motivação.....	15
1.2 Objetivo.....	15
1.3 Justificativa .....	16
1.4 Organização do trabalho.....	16
2 FUNDAMENTAÇÃO TEÓRICA .....	17
2.1 Tipos de deficiência motora .....	17
2.1.1 Paralisia Cerebral .....	18
2.1.2 Paraplegia ou Tetraplegia.....	19
2.1.3 Hemiplegia e Hemiparesia.....	20
2.1.4 Triplegia e Triparesia .....	21
2.1.5 Monoplegia e Mono paresia.....	21
2.1.6 Amputação .....	22
2.2 Cérebro e a Neurociência .....	23
2.2.1 Eletroencefalograma - ECG.....	25
2.3 Ondas Cerebrais.....	26
2.4 Leitura dos dados .....	29
2.4.1 Leitores de sinais cerebrais .....	29
2.4.1.1 Eletroencefalograma - EEG .....	29
2.4.1.2 Eletrocorticografia - ECoG .....	32
2.5 Processamento dos dados com Rede Neural Artificial .....	32
2.5.1 Neurônio Artificial .....	32
2.5.2 Rede <i>Perceptron</i> Multicamadas (MLP).....	35
2.6 Anatomia do antebraço e da mão .....	35
2.6.1 Antebraço .....	35
2.6.2 Mão .....	39
2.7 Atuadores do braço robótico .....	41
2.7.1 Protótipo do braço robótico utilizando impressora 3D .....	42

2.7.1.1 Modelo do braço robótico .....	44
2.7.2 Motores .....	45
3 DESENVOLVIMENTO.....	47
3.1 Metodologia do projeto .....	47
3.1.1 Leitura dos dados.....	48
3.1.2 Processamento dos sinais .....	51
3.1.3 Atuação no braço robótico .....	53
3.2 Implementação e testes .....	55
3.2.1 Junção das etapas do projeto .....	57
3.2.1.1 Leitura e processamento .....	58
3.2.1.2 Processamento e atuador.....	59
3.2.1.3 Leitura, processamento e atuação.....	60
4. RESULTADOS E DISCUSSÕES.....	63
4.1 Primeiro módulo do projeto .....	63
4.2 Segundo módulo do projeto .....	64
5. CONSIDERAÇÕES FINAIS .....	67
5.1 Conclusão.....	67
5.2 Trabalhos futuros .....	67
6. REFERÊNCIAS.....	69
ANEXO A – CÓDIGO DO ARDUINO .....	73
ANEXO B – <i>DATASHEET</i> SENSOR <i>FLEX</i> .....	74
ANEXO C – CÓDIGO DE COLETA NO RASPBERRY .....	76
ANEXO D – CÓDIGO DA RNA .....	79
ANEXO E – <i>DATASHEET</i> SERVO MOTOR MG995.....	81
ANEXO F – CÓDIGO DE ATUAÇÃO DOS MOTORES .....	83

# 1 INTRODUÇÃO

Segundo a pesquisa do Censo 2010, aponta que 45,6 milhões de pessoas declararam possuir algum tipo de deficiência, valor que corresponde mais de 23% da população brasileira. A deficiência visual foi a que teve maior índice 18,6%, a deficiência motora obteve 7%. Com o intuito de buscar ajudar pessoas com deficiência motora, se iniciou o estudo de um dispositivo que obtivesse os sinais cerebrais de um movimento, interpretasse e realiza-se mesmo movimento que o indivíduo pensou.

O cérebro humano é o principal órgão do sistema nervoso central, responsável por comandar o nosso corpo, é nele que se inicia uma ação ou movimento, que são transmitidos através de pulsos elétricos.

Estes pulsos enviados pelo cérebro são denominados de ondas cerebrais, que podem ser caracterizadas pela sua frequência e amplitude. As ondas cerebrais são classificadas em cinco tipos principais, são elas: Alpha, Beta, Delta, Gamma e Theta.

A entrada da RNA foram os sinais recebidos do cérebro em tempo real; como etapa de treinamento está o *dataset* (conjunto de dados gravados a partir dos movimentos determinados), que realizará a classificação entre os dados do *dataset* de acordo com o sinal recebido; em sua saída será a representação do movimento que o indivíduo pensou em realizar, este sinal será o de ativação para a prótese do braço, que fará o movimento que corresponde do sinal recebido na entrada.

A RNA será desenvolvida através do Raspberry Pi 3, que se trata de um minicomputador do tamanho de um cartão de crédito, mas com capacidade e velocidade elevada de processamento, justamente o que é preciso para fazer um dispositivo que consiga ler o cérebro, processar e executar uma ação no final.

A ação final será realizada por uma prótese do braço direito, que foi construída através de uma impressora 3D, utilizando como material o PLA - Filamento de Ácido Polilático derivado a partir do milho e de outros amidos renováveis; indicado para peças que serão expostas á esforços de abrasão. O movimento dos dedos da prótese se dá a partir de uma linha presa no eixo de servos motores que também estão sendo controlados pelo Raspberry.

## 1.1 Motivação

A motivação que levou a realização deste projeto é proporcionar um possível dispositivo que possa auxiliar pessoas que apesar de terem nascido com um dos membros superiores por alguma razão vieram a perder o mesmo.

Segundo o Censo 2010, divulgado pelo IBGE em 2012, aponta que 45,6 milhões de pessoas declararam possuir algum tipo de deficiência, isto corresponde mais de 23% da população brasileira no período da pesquisa; claro que estes dados são referentes a todos os tipos de deficiências, porém é notável que este número é bem alto, por isso o projeto tem como objetivo se tornar uma base para o possível desenvolvimento de um dispositivo capaz de ajudar uma parte deste grupo de pessoas.

## 1.2 Objetivo

O projeto “Dispositivo de Interpretação de Ondas Cerebrais para Controle de um Braço Robótico” teve como objetivo coletar os pulsos elétricos emitidos pelo cérebro humano, processar estes sinais e transforma-los em uma ação de um braço robótico (prótese).

As coletas destes sinais são realizadas através do capacete do dispositivo *MindFlex* (da empresa Neurosky), conectado ao Raspberry Pi 3. O mesmo é um microcomputador, a utilização deste se dá por possuir um melhor tempo de resposta entre o recebimento e o envio de dados.

Com os sinais coletados foi possível envia-los para uma rede neural, que possui a capacidade de interpretar o que o usuário queira fazer (dentro dos movimentos já predefinidos no período de testes). Na saída desta rede neural está a prótese, que se torna o “espelho” do que o cérebro do indivíduo gostaria de fazer com o seu membro, repedindo movimento que já estão presentes na rede.

### **1.3 Justificativa**

O equipamento se tornará capaz de realizar a leitura dos pulsos elétricos do cérebro humano e processar estes sinais através de uma rede neural, para poder transforma-los em uma atividade/ação. Esta atividade ou ação, no caso deste projeto se tornar uma prótese do braço direito, onde poderá representar o membro de uma pessoa amputada.

### **1.4 Organização do trabalho**

Este trabalho foi organizado da seguinte maneira, no capítulo 2, foram informados todos os conceitos que foram necessários como fundamentação teórica para a realização do projeto.

Já no capítulo 3, a metodologia de como foi desenvolvido o projeto a partir das referências existem a respeito, assim como também os testes e resultados emitidos pelo desenvolvimento do projeto.

No capítulo4, os resultados obtidos com a metodologia proposta e o trabalho desenvolvidos são sumarizados e discutidos.

Já no capítulo 5, as conclusões do trabalho são apresentadas, bem como as sugestões de trabalhos futuros para melhorias no projeto.

No capítulo 6, as referências utilizadas para elaboração do trabalho são apresentadas. No final foram apresentados os apêndices do trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

Ao longo deste capítulo foram descritos alguns estudos e pesquisas que estão correlacionados com o tema deste projeto. As principais bases de estudos foram consultadas e os trabalhos mais relevantes utilizados na elaboração deste texto.

### 2.1 Tipos de deficiência motora

A deficiência motora, também conhecida como deficiência física, se trata de uma limitação física ou motora, ou seja, deficiência que afeta na mobilidade do ser humano, normalmente ocorrendo o comprometimento do sistema locomotor, que corresponde aos sistemas osteoarticular, muscular e nervoso.

De acordo com o Decreto nº 5.296 de 02 de dezembro de 2004, deficiência física é:

“Alteração completa ou parcial de um ou mais segmentos do corpo humano, acarretando o comprometimento da função física, apresentando - sob a forma de paraplegia, tetraparesia, triplegia, tri paresia, hemiplegia, hemiparesia, ostomia, amputação ou ausência de membro, paralisia cerebral, nanismo, membros com deformidade congênita ou adquirida, exceto as deformidades estéticas e as que não produzam dificuldade para o desempenho de funções”.

A deficiência motora pode ser classificada com diferentes tipos:

- Temporária – após um determinado tempo de tratamento, o indivíduo volta a apresentar o movimento a área que ocorreu o trauma;
- Permanente – mesmo realizando o tratamento não seria possível voltar as condições de movimento do membro;
- Compensável – permite substituir o membro que sofreu amputação, por exemplo colocando uma prótese.

Já as causas dessa deficiência podem ser classificadas como: congênitas (o indivíduo já nasce com a deficiência) e a adquirida (o indivíduo desenvolve com o tempo).

Existem vários graus de perda de movimentos que podem ser classificados como deficiência motora.

### 2.1.1 Paralisia Cerebral

A paralisia cerebral é uma lesão de uma ou mais áreas do cérebro, que resulta em na diminuição do controle muscular ou até mesmo a paralisia do corpo. As principais causas do desenvolvimento desta deficiência são malformação genética durante a gestação, traumatismo craniano e infecções. (Santana; Filho; Almeida, 2016)

Existem quatro tipos de paralisia:

- Atáxica – indivíduo com dificuldade em suas funções motoras;
- Discinética – movimentos executados involuntariamente pelo indivíduo, o mesmo não controla seus movimentos devido ao excesso de pulsos enviados pelo cérebro;
- Espástica – diminui a força muscular e aumenta as tensões musculares;
- Mista – indivíduo apresenta dois ou mais tipos de paralisia relacionados ao movimento.

Os tratamentos ainda não fazem com que o indivíduo volte a apresentar todos os movimentos, mas podem ajudar a melhorar a postura (fisioterapia) e controlar os distúrbios afetivos e as agitações (medicamentos anticonvulsivantes).

Uma tecnologia que podem ajudar estes indivíduos, é as Interfaces Cérebro Computador. Nota-se um estudo realizado pela equipe de Mestrado em Computação Aplicada, coordenada pelo professor Alejandro Ramirez da Fundação Catarinense de Educação Especial (FCEE) e da Universidade do Vale do Itajaí (Univali), desde 2012, na qual estão desenvolvendo um sistema que possa ajudar pessoas com paralisia cerebral. Formado por um *hardware* em formato de fone de ouvido com sensores na região frontal do cérebro e um *software* que pode ser configurado de acordo com as necessidades do indivíduo (Figura 2.1). Os sensores captam expressões (piscar de olhos ou levantar a sobrancelha) e o *software* processa estas informações transformando em palavras e frases. (Ramirez, 2016)

Figura 2. 1 - Sistema Interfaces Cérebro Computador da FCEE.



Fonte: <http://www.fcee.sc.gov.br/images/NOTICIAS/2016/DEZ/Pesquisa-pioneira-para-pessoas-com-paralisia-cerebral-realizada-pela-FCEE-em-parceria-com-Univali.JPG>.

### 2.1.2 Paraplegia ou Tetraplegia

A paraplegia, ou também conhecida como tetraplegia, é o termo usado para definir indivíduos que não conseguem movimentar os membros superiores ou inferiores, geralmente causado por uma lesão nas vias motoras e sensitivas que transitam até a medula espinhal. (AbcMed, 2013)

As causas mais comuns são os acidentes automobilísticos, mergulho em águas profundas e de forma congênita, que surge devido a lesões cerebrais, decorrentes a gestação ou durante o parto.

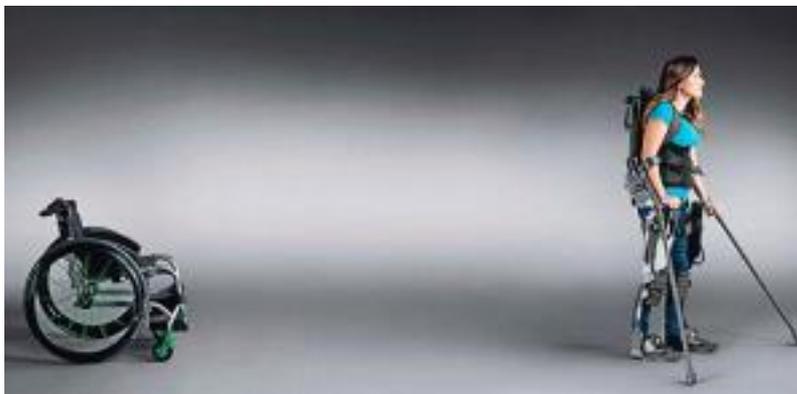
Existem as paraplegias reversíveis e as irreversíveis:

- Reversíveis: são casos de indivíduos que apresentam uma compressão na medula espinhal; onde através de uma cirurgia será possível descomprimir a área afetada, possibilitando novamente a circulação de impulsos nervosos;
- Irreversíveis: quando não mais possibilidade de reverter esta doença, o indivíduo acaba perdendo o movimento dos seus membros.

Indivíduos com paraplegia irreversíveis dos membros inferiores utilizam como suporte de locomoção a careira de rodas, que podem ser movidas manualmente ou eletronicamente pelo ocupante ou empurrada por um terceiro. Mas com o avanço da tecnologia, está se tornando possível um paraplégico começar a ficar de pé e até mesmo caminhar. (Ferreira, 2009)

A empresa americana Ekso Bionics, por exemplo, que criou uma espécie de roupa robô, na qual o indivíduo “veste” o equipamento e consegue ficar de pé e até andar com ajuda de muletas (Figura 2.2). A estrutura é feita de alumínio e titânio, funcionando com uma bateria local, seu peso é de 25 quilos.

Figura 2. 2 - Robô para ser “vestido” por pessoas paraplégicas.



Fonte: <https://cuidarmais.files.wordpress.com/2013/11/eksosuit6.jpg?w=300&h=147>.

### 2.1.3 Hemiplegia e Hemiparesia

A hemiparesia é a diminuição da função motora ou uma paralisia parcial em um dos hemisférios (esquerdo ou direito) do corpo. A hemiplegia é mais grave, conseqüentemente, do que a hemiparesia, uma vez que o indivíduo não apresenta mais movimento em dos lados do corpo, podendo afetar outras funções do corpo, como fala, visão, audição e até a capacidade de raciocínio.

A hemiplegia e a hemiparesia podem ser desenvolvidas com complicações durante a gravidez, acidente vascular cerebral, infecção, agravamentos do quadro diabético e tumores em tecidos cerebrais.

Infelizmente, esta deficiência não é reversível, mas existem tratamentos - como a fisioterapia, hidroterapia e a cirurgia que consiste em um corte de alguns ligamentos desses membros para poder aliviar as contrações musculares – que proporcionam a recuperação parcial dos movimentos. (Coelho, 2017)

### 2.1.4 Triplegia e Triparesia

A triplegia é a perda total das funções motoras em três membros, enquanto a triparesia é a paralisia parcial das funções motoras em três membros. Não existe um padrão nesta deficiência, o indivíduo pode desenvolver a paralisia em ambas as pernas e um braço, ou em ambos os braços e apenas em uma perna. O indivíduo que desenvolve triparesia, não possui os membros completamente paralisados, mas os mesmos são muito fracos. Seu desenvolvimento se dá por paralisia cerebral e até mesmo acidentes vasculares cerebrais.

### 2.1.5 Monoplegia e Mono paresia

A monoplegia é a perda total da função motora de um dos membros superiores ou inferiores; já a monoparesia é a perda parcial das partes motoras de um dos membros superiores ou inferiores. Ambos são causados por lesão do sistema nervoso.

O indivíduo pode desenvolver esta deficiência através de algum acidente vascular cerebral, traumatismo craniano ou até mesmo a má formação congênita durante a gestação.

Uma monoplegia pode evoluir para uma monoparesia, isso pode ocorrer através de um tratamento, como fisioterapia relaxantes musculares, anti-inflamatórios, próteses e entre outros.

Alguns exemplos de equipamentos adaptados para indivíduos com paralisia:

- As pulseiras de pesos são utilizadas para auxiliar nas tarefas diárias, indivíduos com paralisia cerebral apresentam movimentos involuntários nos membros afetados, a pulseira auxilia na redução desses movimentos;
- Teclado adaptado possui setas que podem substituir o *mouse*, se adapta ao teclado tradicional (Figura 2.3) e outras ferramentas.

Figura 2. 3 - Teclado adaptado.



Fonte: <http://intervox.nce.ufrj.br/microfenix/tecladap.jpg>.

### 2.1.6 Amputação

A amputação é a remoção de uma das partes motoras do corpo, geralmente o braço ou a perna. A amputação é realizada quando o membro do indivíduo foi afetado por gangrena (morte do tecido de alguma parte do corpo devido a insuficiência de irrigação sanguínea), quando o membro apresenta algum risco de vida para a saúde do indivíduo (exemplo o câncer) e quando o membro sofreu trauma grave (exemplo um acidente em uma máquina). (Teles, 2013)

Nos outros tipos de deficiência motora, o indivíduo ainda possui o membro, mesmo ele estando paralisado, podendo haver uma melhora no quadro. Já a amputação, infelizmente, deixa o indivíduo sem o membro, não havendo uma melhora no quadro do paciente, este recurso é um dos últimos a serem usados pelos especialistas, pois uma vez amputado o membro, não teria mais como voltar com o mesmo, apenas com a utilização de uma prótese.

Quando um membro é amputado, um braço por exemplo, são ligados as artérias e veias para evitar uma hemorragia, em seguida os músculos são retirados e o osso serrado para finalizar a amputação. A região onde sofreu a amputação (conhecida como coto) é ajustada para que no futuro venha a ser colocado uma prótese no local do trauma, com o objetivo de repor o membro. (Teles, 2013)

Alguns indivíduos que sofreram amputação podem desenvolver um membro fantasma, ou seja, sente-se como se o membro ainda estivesse em seu corpo, podendo sentir dor e sensações.

“Existe um conflito entre a perda anatômica, a imagem corporal cerebral, a memória sensitiva e aspectos psíquicos. O membro sai do corpo, mas não sai do crebro, isso pode gerar o aparecimento do membro fantasma, algum desconforto e até dores crônicas de difícil tratamento”. (Teles, 2013)

No cérebro humano possui uma espécie de mapa corporal, onde cada parte do corpo está registrada. Apesar do cérebro visualizar que o indivíduo não apresenta mais o membro, ele ainda pode enviar pulsos para o local, pois se a região onde sofreu a amputação apresentar sinais musculares ou do sistema nervoso o cérebro entende que ainda pode existir a continuação do membro.

## 2.2 Cérebro e a Neurociência

O cérebro é um dos órgãos de maior importância do sistema nervoso do corpo humano pois é nele que são comandadas as ações motoras, a comunicação dos estímulos sensoriais e as atividades neurológicas.

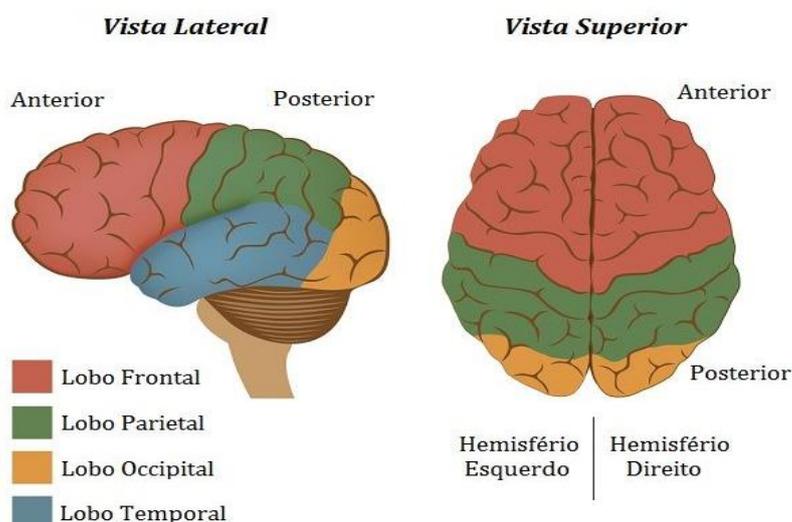
Formado por dois tipos de tecidos, o córtex cerebral e o núcleo cerebral:

- Córtex cerebral: de coloração cinza, é responsável por funções de processamento de informações e a linguagem, está composta pelas células dos neurônios e outras células do sistema nervoso;
- Núcleo cerebral: de coloração branca, responsável pela comunicação entre o córtex cerebral, os órgãos sensoriais e os músculos do corpo.

O tecido córtex cerebral é subdividido em lobos cerebrais, que podem ser classificados em quatro categorias como apresentado na Figura 2.4.

- Lobo frontal: responsável pelo pensamento, planejamento e emoção;
- Lobo parietal: responsável pela sensação de dor, tato, temperatura e pressão;
- Lobo temporal: relacionado com o sentido de audição, processamento da memória e emoção;
- Lobo occipital: processamento de informações visuais.

Figura 2. 4 - Lobos Cerebrais.



Fonte: <https://static.todamateria.com.br/upload/56/78/56784e94abd35-cerebro.jpg>.

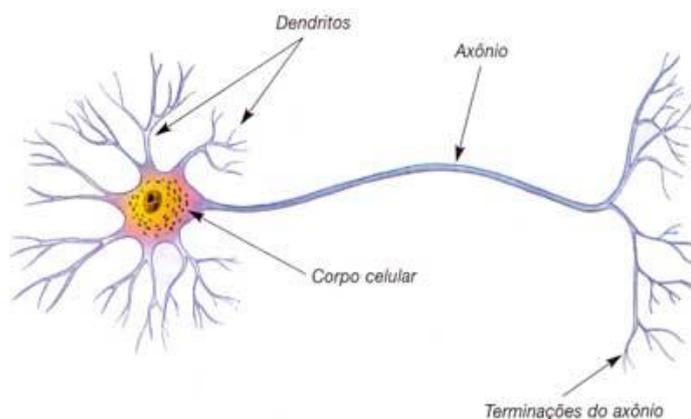
O cérebro em si pode ser dividido em duas partes o hemisfério direito e o hemisfério esquerdo. O hemisfério direito é responsável pela consciência artística, criatividade, intuição, imaginação e quem possui o este hemisfério dominante será uma pessoa de escrita canhota (ou seja, irá escrever com a mão esquerda). Já o hemisfério esquerdo é responsável por pensamentos analíticos, raciocínio, linguagem, lógica, escrita e quem possui este hemisfério dominante será uma pessoa destro (ou seja, que escreve com a mão direita).

A estrutura fundamental do cérebro para processamentos das informações e estímulos do corpo é o neurônio. Os neurônios são células nervosas, responsáveis por conduzir os impulsos nervosos. Um ser humano adulto pode possuir cerca de 86 bilhões de neurônios.

Os neurônios podem ser divididos em três partes:

- Axônios: envia os sinais elétricos para outros neurônios, ou de um neurônio para uma glândula ou fibra muscular;
- Corpo celular: região onde ocorre o processamento das informações recebidas;
- Dendritos: local onde o neurônio recebe os sinais elétricos enviados por outros neurônios.

Figura 2. 5 - Estrutura de um neurônio.



Fonte: <http://www.sogab.com.br/anatomia/neuronio.jpg>.

Para que o cérebro realize alguma ação é necessário que o mesmo emita sinais elétricos para a execução do pensamento do indivíduo, estes sinais são denominados como ondas cerebrais.

### 2.2.1 Eletroencefalograma - ECG

As ondas cerebrais estão sendo estudadas desde 1929, quando o psiquiatra alemão chamado Hans Berger anunciou que era possível registrar pequenos sinais de correntes elétricas gerados a partir do cérebro humano, não precisando abrir o crânio do indivíduo para colocar os eletrodos. Berger observou que estes sinais mudavam de acordo com o estado do indivíduo, durante o sono, na anestesia, na ausência de oxigênio e em certas doenças nervosas.

Os sinais recebidos pelo experimento eram registrados em papéis, Berger deu o nome a esta leitura dos sinais cerebrais como eletroencefalograma, ou EEG. (Sabbatini, 1998)

Figura 2. 6 - Sinais registrados por Hans Berger.



Fonte: <http://www.cerebromente.org.br/n03/tecnologia/1st-ecg.gif>.

Berger (2015), realizou seus experimentos primeiro em pessoas com defeitos no crânio, depois de alguns anos, com o equipamento um pouco mais sensível na realização da leitura, começou a estudar indivíduos sem nenhuma doença.

A princípio, Berger chamou a frequência dominante de 'baixa frequência de primeira ordem' e as outras de 'alta frequência de segunda ordem'. Depois por um tempo, optou por umas nomenclaturas mais simples, as ondas de primeira ordem denominaram como 'alpha –  $\alpha$ ', já as de segunda ordem com o nome de 'beta –  $\beta$ '.

Mais tarde, em 1958, Joe Kamiya iniciou uma nova linha de pesquisas com o objetivo de saber se o ser humano teria a capacidade de controlar os sinais enviados pelo cérebro.

Os experimentos eram feitos em duas partes. A primeira era pedida ao indivíduo que fechasse os olhos e ouvisse o som que era colocado, este som era o com frequência alpha, então era perguntado qual o som que o indivíduo estava ouvindo e informado se ele estava certo ou errado. Na segunda parte, o indivíduo tinha que entrar no estado alpha ao ouvir o som de uma campainha. Assim podendo descobrir que o estado alpha está relacionado ao relaxamento. (Gomes, 2015)

Assim foram iniciados os estudos sobre os sinais do cérebro, encontrando as frequências principais do funcionamento do cérebro: Gamma, Beta, Alpha, Theta e Delta.

### **2.3 Ondas Cerebrais**

As ondas cerebrais são classificadas como pulsos elétricos que se espalham e forma a comunicação entre a mente, o cérebro e o corpo. Podem ser classificadas por frequência dos pulsos elétricos, que geralmente é medido em Hertz (Hz), onde equivale a um ciclo por segundo. (França, 2008)

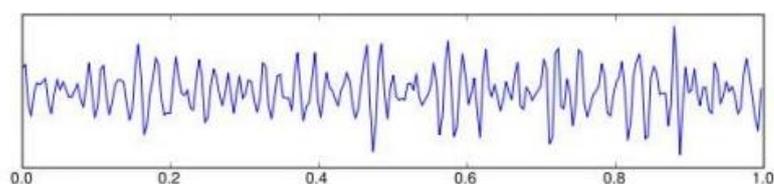
Os cinco tipos de ondas cerebrais são: Gamma, Beta, Alpha, Theta e Delta.

- Ondas Gamma

Chegando a ser definida entre 30 e 70Hz de frequência, a onda gamma é a maior entre os outros sinais enviados pelo cérebro.

Esta onda sempre está presente na mente humana, até mesmo quando estamos dormindo. Está relacionada ao processamento de estímulos visuais, táteis e auditivos. A Figura 2.7 mostra o comportamento de uma onda gamma. (França, 2008)

Figura 2. 7 - Onda Gamma.



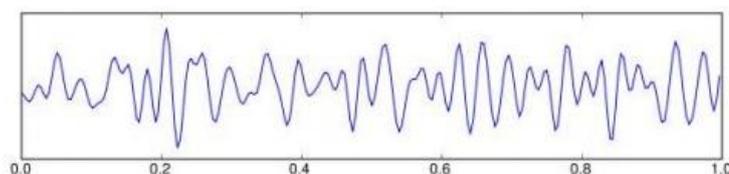
Fonte: FRANÇA, Rafael Ferreira. Dissertação: Indutor de ondas cerebrais por batimento binaural. Ano de 2015.

- Ondas Beta

Chegando a ser definida entre 13 e 30Hz de frequência. Sempre que realizar a leitura de um sinal Beta, geralmente coexiste de obter também um sinal Gamma.

Está associado a emoções como medo, raiva, ansiedade, atenção e concentração, geralmente apresentado durante o dia, que é quando estamos no estado de “vigília” física. A Figura 2.8 mostra o comportamento de uma onda beta. (França, 2008)

Figura 2. 8 - Onda Beta.

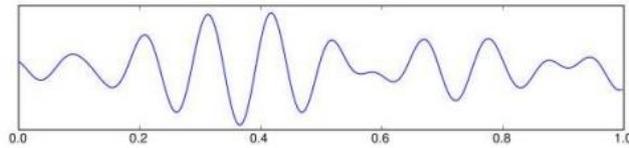


Fonte: FRANÇA, Rafael Ferreira. Dissertação: Indutor de ondas cerebrais por batimento binaural. Ano de 2015.

- Ondas Alpha

Está definida entre 8 e 13Hz de frequência. Ondas alpha são a conexão entre a comunicação do consciente (ondas Beta) e o inconsciente (ondas Theta). Caracterizada por momentos tranquilos, relaxamento físico e mental. (França, 2008)

Figura 2. 9 - Onda Alpha.

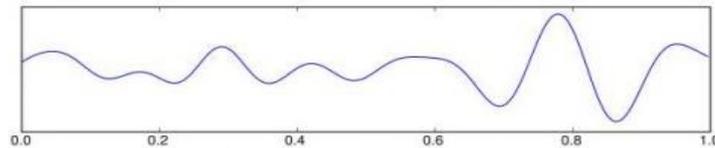


Fonte: FRANÇA, Rafael Ferreira. Dissertação: Indutor de ondas cerebrais por batimento binaural. Ano de 2015.

- Ondas Theta

Chegando a ser definida entre 4 e 8Hz de frequência. Representa o estado de inconsciência, está presente nos sonhos e em meditação profunda; relacionada também com a memória de curto prazo. A Figura 2.10 mostra o comportamento de uma onda theta. (França, 2008)

Figura 2. 10 - Onda Theta.

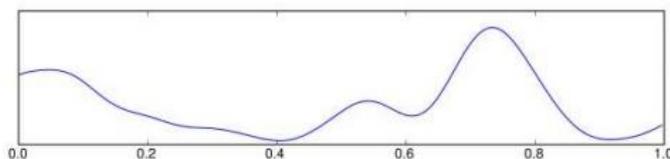


Fonte: FRANÇA, Rafael Ferreira. Dissertação: Indutor de ondas cerebrais por batimento binaural. Ano de 2015.

- Ondas Delta

A mais lenta de todos os sinais do cérebro, a onda delta está entre 1 e 4 Hz de frequência. Presente durante o sono profundo e até mesmo no coma. A Figura 2.11 mostra o comportamento de uma onda delta. (França, 2008)

Figura 2. 11 - Onda Delta.



Fonte: FRANÇA, Rafael Ferreira. Dissertação: Indutor de ondas cerebrais por batimento binaural. Ano de 2015.

## **2.4 Leitura dos dados**

O cérebro humano pode emitir sinais elétricos a cada pensamento, ação ou sentimento, sinais estes que podem ser medidos através de eletrodos colocados na cabeça do indivíduo.

Realizando a leitura dos sinais é possível convertê-los em gráficos com os determinados valores que corresponde a cada onda cerebral, assim podendo analisar o comportamento do cérebro em determinados movimentos.

Para poder interpretar os sinais, é preciso de uma excelente análise dos dados de teste e um processamento bem executado. Assim, no final desta fase os resultados podem ser aplicados em uma prótese para pessoas com deficiência motora ou ajudar indivíduos que fazem tratamento de fisioterapia ou de recuperação dos movimentos.

### **2.4.1 Leitores de sinais cerebrais**

Alguns dos métodos utilizados para realizar a leitura dos sinais do cérebro são o Eletroencefalograma (EEG) e o Eletrocorticografia (ECoG).

#### **2.4.1.1 Eletroencefalograma - EEG**

O eletroencefalograma (EEG) ou também conhecido como eletroencefalografia, possibilita registrar os sinais elétricos do cérebro através de eletrodos colocados no couro cabeludo. Começou a ser utilizado em 1924, quando o alemão Hans Berger descobriu que o cérebro humano poderia gerar uma atividade elétrica capaz de ser registrada.

A partir de então, o EEG começou a evoluir e hoje se trata de um método não invasivo que registra as oscilações neurais do cérebro, ou seja, as ondas cerebrais.

No mercado existem diversos modelos de leitor para EEG, o que diferencia de um modelo para outro é o número de sensor existentes. Podendo ter modelos com um, cinco, quatorze, dezenove ou até mais sensores.

A Figura 2.12 mostra o modelo de EEG da empresa EPOC Emotiv, na qual possui 14 sensores que realizam a leitura do cérebro, funcionando como se fosse um capacete.

Figura 2. 12 - Leitor EEG da empresa EPOC Emotiv.



Fonte: [https://www.emotiv.com/wp-content/uploads/2016/06/emotiv\\_epoc\\_square-w.jpg](https://www.emotiv.com/wp-content/uploads/2016/06/emotiv_epoc_square-w.jpg).

Já a Figura 2.13 mostra o capacete do dispositivo *MindFlex*, da empresa Neurosky, no qual possui um sensor externo (não invasivo). Foi desenvolvido para o controle de um brinquedo, onde o jogador tinha que se concentrar para poder controlar a altura de uma esfera sobre uma mesa do jogo.

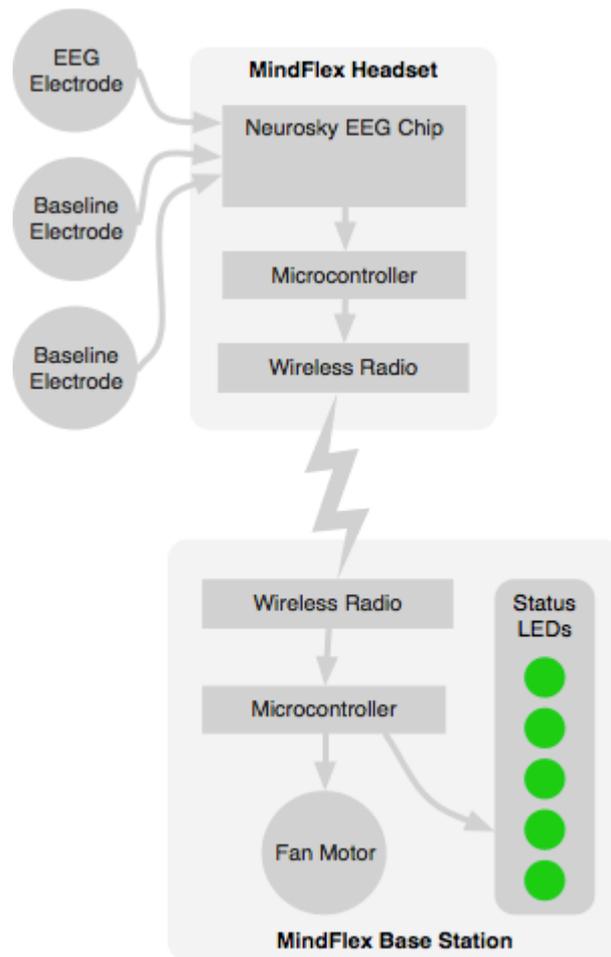
Figura 2. 13 - Leitor EEG do dispositivo *MindFlex*.



Fonte: <http://d2ydh70d4b5xgv.cloudfront.net/images/9/8/mattel-mind-flex-and-dual-games-replacement-headset-mindflex-game-d3e4377cd293b9ae2ab08482557f8605.jpg>.

O *layout* básico do *hardware* do dispositivo *Mindflex* está representado na Figura 2.14. o sensor que realiza a leitura dos sinais do cérebro está localizado na faixa que vai estar em contato com a testa do indivíduo. Um microcontrolador na lateral do capacete analisa os dados do sensor EEG e envia via transmissão de rádio para a mesa do dispositivo, onde o ventilador presente na mesa levita a esfera e acende os leds de acordo com o nível de atenção do indivíduo. (Vidich; Yuditskaya, 2013)

Figura 2. 14 - *Layout* do *hardware* do dispositivo *MindFlex*.



Fonte: <http://frontiernerds.com/files/mindflex-schematic.gif>.

### **2.4.1.2 Eletrocorticografia - ECoG**

A eletrocorticografia (ECoG) pode registrar atividades elétricas do córtex cerebral com muito mais precisão do que o EEG, porém se trata de um leito invasivo, pois é necessário realizar uma cirurgia para colocar os sensores diretamente em contato com o cérebro do indivíduo. (Longo; Pupo; Lemini; Pazzanese; Pimenta, 1951)

## **2.5 Processamento dos dados com Rede Neural Artificial**

A Rede Neural Artificial (RNA) é uma técnica computacional que tem como base a estrutura neural do cérebro, onde a partir de um período de treinamento é possível adquirir novos conhecimentos e experiência. Após o período de treinamento é possível deixar o sistema agir de forma independente, ou seja, de forma autônoma.

As pesquisas sobre o desenvolvimento das RNAs se iniciaram na década de 50, mas o tema só ganhou espaço mundialmente após os anos de 1990, nas áreas da economia, biologia, ecologia e entre outros. (Silva et.al. – 2010)

As características principais de uma RNA são:

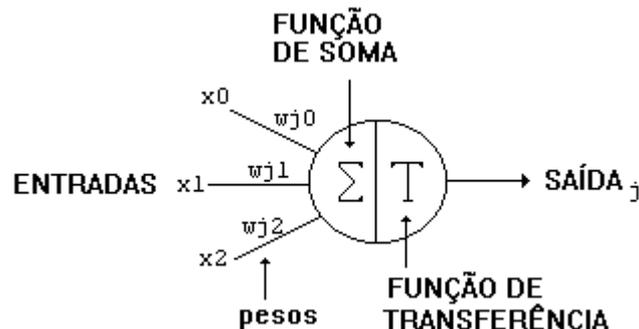
- Adaptação por experiência – as adaptações dos parâmetros da rede se dão em cada experiência que é exposta, assim ajustando seus pesos sinápticos para obter uma melhor resposta em sua saída;
- Capacidade de aprendizado – com o período de treinamento a rede consegue aprender quais serão os seus modos de atuação;
- Habilidade de generalização – após a fase de treinamento a rede é capaz de criar soluções que não foram dadas para a mesma, pois a rede se adapta de acordo com a sua experiência.

### **2.5.1 Neurônio Artificial**

Uma RNA é composta por vários neurônios artificiais, nos quais são estruturas lógicas – matemáticas que se assemelham com o comportamento e as funções de um neurônio biológico.

No neurônio artificial, o dendrito (prolongamento do neurônio biológico, responsável pela recepção do estímulo) foi modificado para “entrada”. A ligação entre o corpo celular artificial e a entrada é realizado através do “peso” (representa a sinapse do sistema biológico).

Figura 2. 15 - Neurônio artificial de uma rede neural.



Fonte: [http://www.cerebromente.org.br/n05/tecnologia/neuronio\\_artificial.gif](http://www.cerebromente.org.br/n05/tecnologia/neuronio_artificial.gif).

A Figura 2.15 mostra um neurônio artificial, onde dispõem os elementos básicos desta estrutura.

Os sinais de entrada ( $x_0, x_1, \dots, x_n$ ) são os valores que entram na rede para serem avaliados e comparados com os dados já colocar na fase de treinamento dentro da rede. (Tafner, 1998)

Os pesos sinápticos ( $w_1, w_2, \dots, w_n$ ) são os valores para ponderar cada valor de entrada da rede, permitindo classificar os valores mais relevantes. (Tafner, 1998)

A função de soma ou combinador linear ( $\Sigma$ ) tem como função somar todos os sinais de entrada que foram ponderados pelos seus pesos, para gerar um valor de potencial de ativação. (Tafner, 1998)

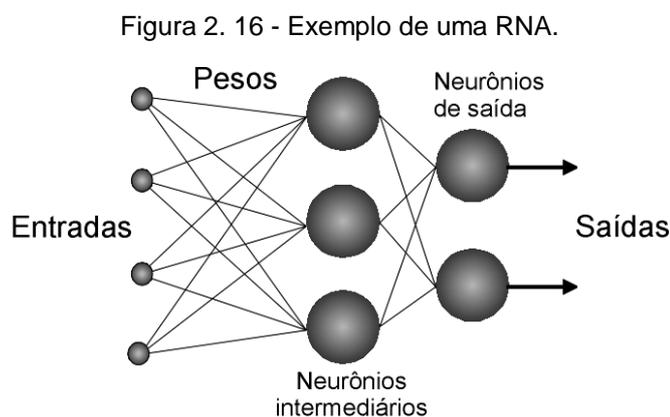
O limiar de ativação ( $\theta$  - bias) é responsável por verificar se o valor da função soma é igual ao bias, se for o mesmo é repassado para a próxima etapa do neurônio, caso contrário o sinal não será transferido. (UFSC, 2012)

A função de transferência é o processo de verificar se o valor da função soma é igual ao bias. (UFSC, 2012)

Antes da função de transferência possui a função de ativação onde será decidido o que fazer com o resultado da soma ponderada dos sinais de entrada (ativando ou não). (UFSC, 2012)

O sinal de saída (y) consiste no valor final produzido pelo neurônio com relação aos valores de entrada.

Todas as RNAs possuem os neurônios artificiais em sua estrutura, mas o que diferencia uma RNA de outra são as conexões entre as camadas, as camadas intermediárias, a quantidade de neurônios, a função de transferência e o algoritmo de aprendizado. (Tafner, 1998)



Fonte: <http://www.cerebromente.org.br/n05/tecnologia/image11.gif>.

O algoritmo de aprendizado é uma das partes mais importantes de uma RNA, onde deverá ter a capacidade de aprender com a sua rotina e com isso pode melhorar o seu desempenho. Tratasse de um conjunto de regras que são definidas na sua criação, com o objetivo de problemas de aprendizado ou treinamento. O que diferencia um algoritmo de aprendizado de outro é o modo como os pesos são alterados.

A partir dos sinais do cérebro e dos padrões encontrados, tornasse possível a criação de uma RNA. Na qual, os sinais coletados pelo capacete irão se tornar a entrada da rede, os padrões serão os pesos e a saída será vista na movimentação da prótese. (Tafner, 1998)

### **2.5.2 Rede *Perceptron* Multicamadas (MLP)**

A rede MLP (*Multi – layer Perceptron*) é composta por camadas de neurônios ligadas entre si por sinapses com pesos. Possui todas as conexões entre seus neurônios, ou seja, um neurônio em qualquer camada da rede estará conectado a todos os outros neurônios da camada anterior. O fluxo de sinais da rede MLP é sempre feito da esquerda para a direita, camada por camada. (Ramos, 2003)

No treinamento, a rede opera em uma sequência de passos com o algoritmo de retro programação (*backpropagation*). Em um primeiro momento, um sinal é inserido na camada de entrada da rede, o resultado flui de camada para camada até chegar na saída da rede. No segundo momento, a saída obtida é comparada à saída desejada para o sinal inserido no início do processo; se a saída obtida comparada com a saída desejada não estiver correta, o erro é calculado. O erro é programado a partir da camada de saída até a de entrada, e os pesos das conexões existentes ao longo da rede vão sendo modificados conforme o erro, ajustando para que a saída coincida com o esperado. (Ramos, 2003)

## **2.6 Anatomia do antebraço e da mão**

A anatomia humana consiste em estudar detalhadamente o corpo humano, possibilitando o diagnóstico e tratamento médico mais eficazes no combate a diversas doenças.

### **2.6.1 Antebraço**

Na anatomia humana, campo da biologia responsável por estudar a forma e a estrutura do organismo humano, dá o nome de antebraço a região entre o cotovelo e o punho (ou carpo). Os músculos (tecidos responsáveis pelos movimentos do indivíduo) do antebraço podem ser classificados em três grupos: anterior – com 8 músculos; lateral – com 4 músculos e posterior – com 8 músculos. (Wecker, 2001)

- Anterior
  - Pronador redondo: está localizado no plano superficial da região anterior do antebraço; responsável por flexão do cotovelo e rotação do antebraço;

- Flexor radial do carpo: sua localização está entre o pronador redondo e o palmar longo; responsável por flexão do cotovelo, rotação do antebraço e abdução da mão;
- Palmar longo: situado na parte superficial da face anterior do antebraço; responsável pela flexão palmar;
- Flexor ulnar do carpo: localizado na parte antero-lateral de todo o antebraço;
- Flexor superior dos dedos: localizado na segunda camada muscular do antebraço; encarregado pela flexão, abdução e condução dos dedos;
- Flexor profundo dos dedos: está coberto pelo músculo superficial dos dedos; sua ação é influenciada na flexão palmar e adução da mão;
- Flexor longo do polegar: sua localização é a mesma que o flexor profundo dos dedos; responsável por flexão palmar e adução da mão;
- Pronador quadrado: está na região mais profunda em relação aos outros músculos do antebraço; sua ação é agir na pronação (inclinação).

(Wecker, 2001)

Figura 2. 17 - Músculos do antebraço vista anterior.



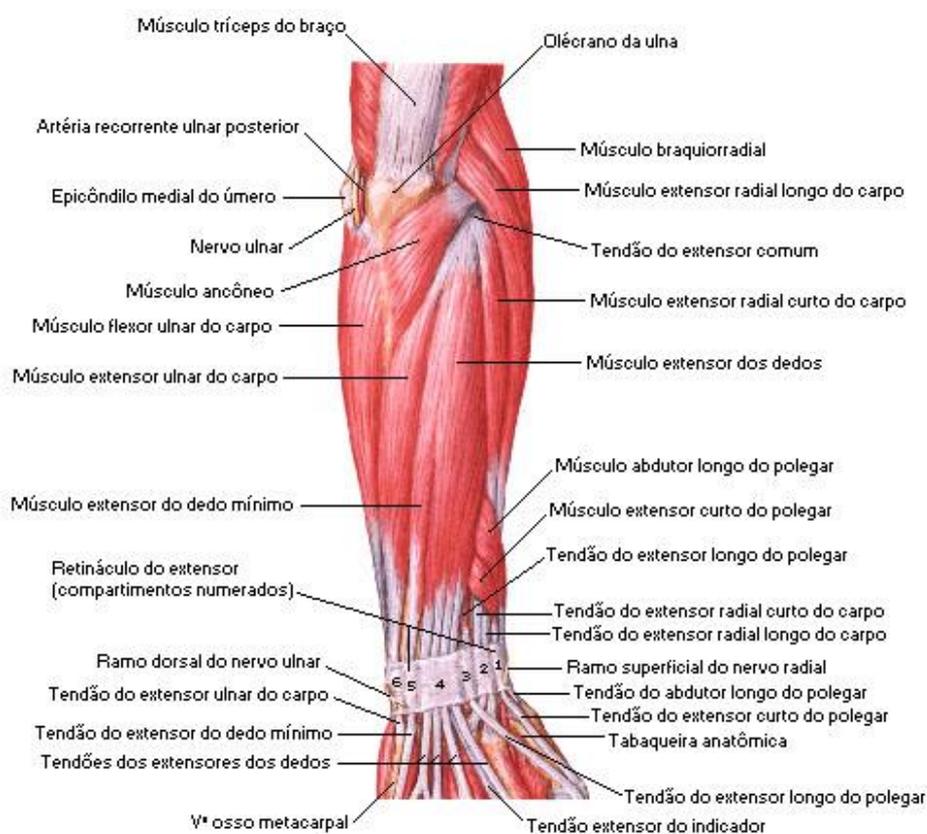
Fonte: [https://www.auladeanatomia.com/upload/site\\_pagina/antebraço2.jpg?x73185](https://www.auladeanatomia.com/upload/site_pagina/antebraço2.jpg?x73185).

- Lateral

- Braquiorradial: músculo superficial da região lateral do antebraço; ação de flexão, pronação e rotação da palma da mão para cima;
- Extensor radial longo do carpo: está localizado na parte de baixo do músculo braquiorradial; sua ação são as mesmas que o braquiorradial;
- Extensor radial curto do carpo: localizado na região lateral e posterior do antebraço; responsável pelo dorso flexão e abdução da mão;
- Supinador: está localizado na parte mais profunda da região do antebraço; encarregado pela movimentação da rotação da palma da mão para cima.

(Wecker, 2001)

Figura 2. 18 - Músculos do antebraço vista lateral.

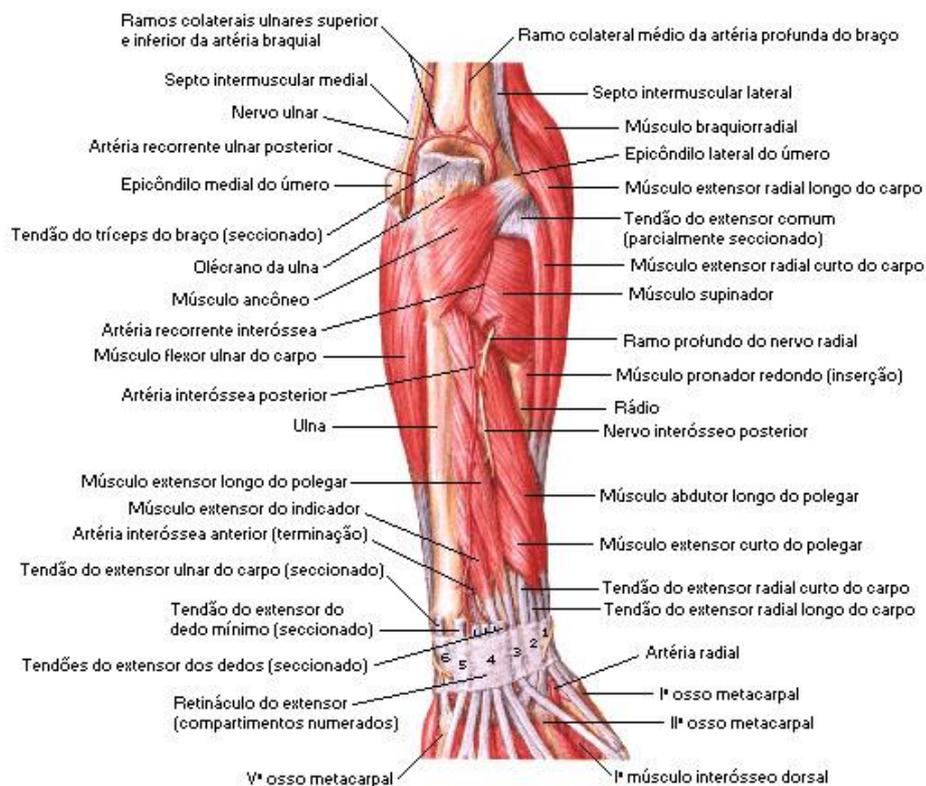


Fonte: [https://www.auladeanatomia.com/upload/site\\_pagina/antebraço.jpg?x73185](https://www.auladeanatomia.com/upload/site_pagina/antebraço.jpg?x73185).

- Posterior
  - Extensor dos dedos: se divide em quatro tendões quando localizado perto do carpo; responsável por extensão e dorso flexão dos dedos;
  - Extensor do 5º dedo (dedo mínimo): está entre o músculo extensor dos dedos; encarregado pela extensão e dorso flexão do dedo mínimo;
  - Extensor ulnar do carpo: está entre o músculo extensor do dedo mínimo; responsável por extensão, dorso flexão e abdução da mão;
  - Ancônio: músculo da parte posterior do cotovelo; responsável pela extensão do antebraço;
  - Abductor longo do polegar: localizado na parte lateral do antebraço; responsável pela ação de abdução do polegar e da mão;
  - Extensor curto do polegar: está localizado junto ao músculo abductor longo do polegar; responsável pela extensão do polegar e abdução da mão;
  - Extensor longo do polegar: está embaixo do músculo extensor dos dedos; encarregado pela abdução, adução e extensão do polegar;
  - Extensor do 2º dedo (dedo indicador): está localizado junto ao músculo extensor do polegar; sua ação é de extensão do indicador.

(Wecker, 2001)

Figura 2. 19 - Músculos do antebraço vista posterior.



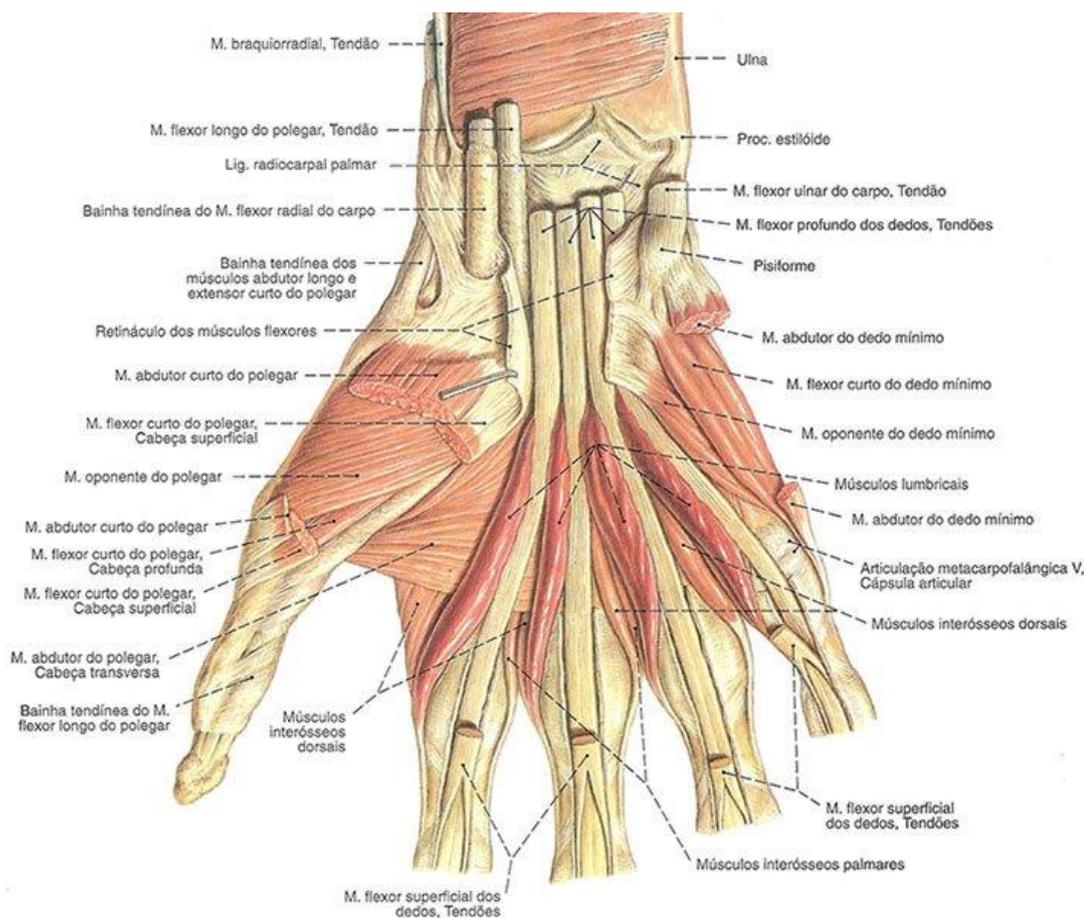
Fonte: [https://www.auladeanatomia.com/upload/site\\_pagina/antebraço1.jpg?x73185](https://www.auladeanatomia.com/upload/site_pagina/antebraço1.jpg?x73185).

## 2.6.2 Mão

A mão é a parte final do membro superior, indo do punho até as pontas dos dedos. Composta por cinco dedos, polegar indicador, médio, anelar ou anular e mínimo. Os dedos indicadores e polegares, são os únicos que podem ser movimentados sem alterar a movimentação dos demais. (Wecker, 2001)

A mão pode ser classificada em três grupos: palmar lateral (tenar), palmar medial (hipotenar) e palmar média.

Figura 2. 20 - Músculos da mão direita.



Fonte: <http://anatomiaonline.com/wp-content/uploads/2015/09/5.9-Superior18.jpg>.

- **Palmar lateral**
  - Abductor curto do polegar: localizado na parte superficial da mão; responsável por abdução e flexão do polegar.
  - Flexor curto do polegar: localizado junto ao abductor curto do polegar; encarregado pela adução e flexão do polegar;
  - Oponente do polegar: localizado embaixo do musculo do abductor curto do polegar; responsável pela adução do polegar;
  - Adutor do polegar: localizado no plano profundo da mão; encarregado pela adução e flexão do polegar.
  
- **Palmar medial**
  - Palmar curto: ação de extensão da pele da palma da mão;

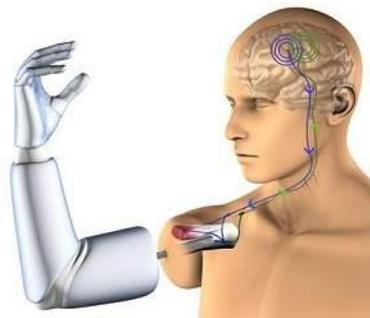
- Abdutor do dedo mínimo: localizado na região hipotênar; responsável pela extensão e abdução do dedo mínimo;
  - Oponente do dedo mínimo: localizado embaixo do musculo abdutor do dedo mínimo.
- **Palmar média**
    - Lumbricais: possui quatro finos feixes musculares; responsável por flexão, abdução e extensão dos dedos;
    - Interósseos palmares: composto por três músculos; localizado na parte profunda da face da palma da mão; responsável por flexão e extensão dos dedos;
    - Interósseos dorsais: possui quatro músculos bipeniformes; responsável pela flexão, abdução e extensão dos dedos.

## 2.7 Atuadores do braço robótico

As próteses são dispositivos artificiais que substituem o membro ausente do indivíduo, que por algum motivo perdeu ou veio a nascer sem o membro. Este dispositivo pode ser dividido em das categorias biônica e mecânica.

- **Biônica:** possibilita a movimentação normal ou quase perfeita do membro amputado; movida com impulsos nervosos do indivíduo;

Figura 2. 21 - Prótese biônica.



Fonte: <http://www.inovacaotecnologica.com.br/noticias/imagens/010180121211-protese-controlada-pensamento-2.jpg>.

- Mecânica: possui funções mais limitadas em relação a prótese biônica, possibilitando apenas movimentos reduzidos; acionadas, por exemplo, por um sistema de molas.

Figura 2. 22 - Prótese mecânica.



Fonte: <https://s-media-cache-ak0.pinimg.com/736x/b3/fd/9f/b3fd9f2243a057f0c3a714d88e56ee6f--in-south-africa-the-two.jpg>.

A prótese pode ser confeccionada com diversos materiais, como titânio, aço ou alumínio. No mercado, as próteses do membro superior desenvolvidas com estes mesmos materiais fabricados na China, Reino Unido, Alemanha ou Estados Unidos podem chegar ao preço de US\$100 mil ou cerca de R\$350 mil (Fonte Revista Exame de 2016). (Peña, 2016)

Atualmente, estão sendo desenvolvidas próteses visando o seu baixo custo, mas que mantenham a sua função em substituir o membro. Estas próteses são produzidas através da impressão em 3D.

### **2.7.1 Protótipo do braço robótico utilizando impressora 3D**

A impressão 3D possui uma tecnologia conhecida como *Fused Deposition Modeling* - Modelagem por Função e Deposito (FDM), onde funciona através da adição de camadas sobrepostas. (Kruk, 2017)

Através de um *software* de edição em três dimensões, é montado a próteses de acordo com as dimensões do indivíduo. Necessário nesta fase do projeto detalhar ao máximo como o objeto ficará em seu resultado, pois assim contribui para que a prótese esteja adequada para o usuário.

O material para a impressão 3D são os filamentos PLA, ABS e PETG.

- PLA – Ácido Polilático – derivado a partir do milho e outros amidos renováveis; filamento de fácil impressão; indicado para peças que serão expostas á esforços de abrasão;
- ABS – Acrilonitrila Butadieno Estireno- derivado do petróleo; resistente à temperatura e absorção de impactos;
- PETG – Politereftalato de Etileno Glicol – derivado também do petróleo; resistente mecanicamente, quimicamente e a temperatura.

(Dorfer, 2015)

O quadro 2.1 mostra algumas características de cada um dos filamentos para impressão.

Quadro 2. 1 - Comparação dos materiais.

Propriedade/Desempenho	Menor	Médio	Maior
Ecológico	ABS	PETG	PLA
Brilho	ABS	PLA	PETG
Transparência	ABS	PLA	PETG
Rigidez/Dureza	PETG	ABS	PLA
Resistência a impactos	PLA	ABS	PETG
Flexibilidade	PLA	ABS	PETG
Contração	PLA	PETG	ABS
Precisão em detalhes	ABS	PETG	PLA
Qualidade de superfície da peça	ABS	PETG	PLA
Resistência a atritos	PLA	ABS	PETG
Resistência química	ABS	PLA	PETG
Resistência altas temperaturas	PLA	PETG	ABS
Facilidade de pós processamentos	PLA	PETG	ABS
Usinabilidade	PLA	PETG	ABS
Densidade	ABS	PLA	PETG
Preço por cm <sup>3</sup> /grama/hora	PLA	ABS	PETG

Fonte: <http://www.impressao3dfacil.com.br/conheca-os-diferentes-tipos-de-materiais-para-impressao-3d-fdm/>.

### 2.7.1.1 Modelo do braço robótico

O material mais usado nas próteses de baixo custo no mercado é o PLA, justamente por não ser composto por petróleo e por ser um material totalmente ecológico e renovável. (Cardoso, 2017)

Existem diversos modelos de próteses para o braço direito com base em impressão 3D. Uma das que apresenta melhor adaptação para o desenvolvimento deste projeto é o robô humanoide InMoov.

Desenvolvido pelo *designer* e escultor francês Gaël Langevin, o InMoov é um robô que se assemelha do corpo humano, da cintura para cima. Todas as suas peças foram impressas em impressora 3D e possuem o tamanho real de um ser humano. O InMoov possui todas as suas peças em código aberto, ou seja, basta entrar no site e baixar os componentes para imprimir.

Figura 2. 23 - Braço esquerdo do InMoov.



Fonte: [http://inmoov.fr/wp-content/uploads/2015/07/inmoov\\_robot\\_arm\\_3d\\_print220-768x1024.jpg](http://inmoov.fr/wp-content/uploads/2015/07/inmoov_robot_arm_3d_print220-768x1024.jpg).

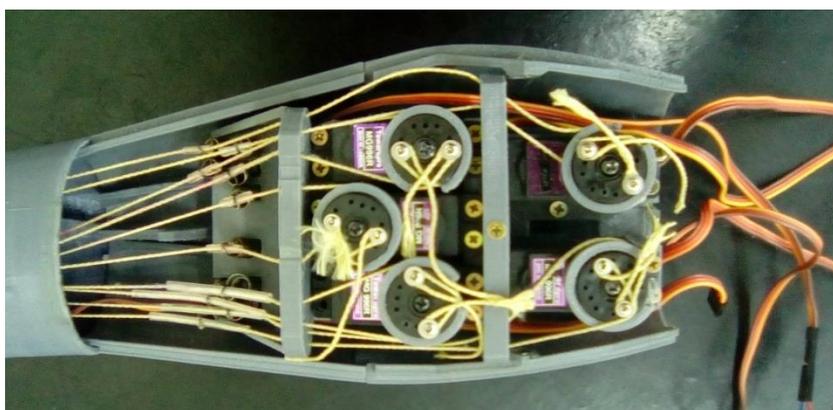
A Figura 2.23 representa o braço esquerdo do robô InMoov já montado, dentro da prótese possui 5 servos motores para o acionamento dos dedos (abrir e fechar dos membros) e 1 servo para a rotação do pulso.

Para o desenvolvimento da prótese deste projeto será utilizado o braço direito do robô InMoov, impresso a partir de filamento PLA. Seu acionamento será via sinais coletados do cérebro que irão passar por um processamento antes de chegar até o atuador (a prótese).

## 2.7.2 Motores

Para a movimentação da prótese é necessário a colocação de motores de pequeno porte e que sejam possíveis controlar a posição que seja necessária. No caso da prótese para o InMoov é indicado o servo motor MG995, se encaixa perfeitamente no braço robótico, como mostrado na Figura 2.24. (Penha; Lacerda, 2013)

Figura 2. 24 - Posição dos motores na prótese.



Fonte: Autor.

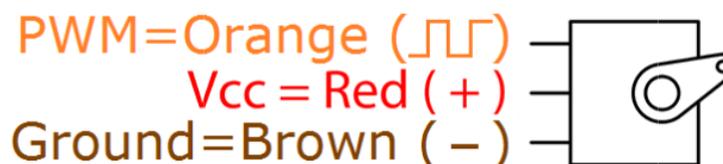
O servo motor é um dispositivo eletromecânico que possui um *encoder* e um controlador acoplado em seu sistema, podendo apresentar movimento rotativo proporcional a um comando de forma que atualize a sua posição. (Penha; Lacerda, 2013)

Para controlar a sua velocidade e posição do eixo o servo utiliza um sinal de feedback. Os servos motores são compostos por:

- Atuador: conjunto de engrenagens e redução no seu interior com objetivo de amplificar o seu torque (movimento giratório). O servo MG995 trabalham com atuadores de até 180° de liberdade de giro;
- Sensor (*encoder*): medidor de posição, normalmente utilizado um potenciômetro acoplado ao eixo do motor, cuja resistência elétrica está associada a posições diferentes do eixo do motor;
- Controlador e circuitos complementares: A saída dos sensores é utilizada como realimentação e comparada com o sinal de controle;

Os servos são acionados através de três fios, dois para alimentação e um que corresponde ao sinal de controle, como mostrado na Figura 2.25.

Figura 2. 25 - Conectores do servo motor.



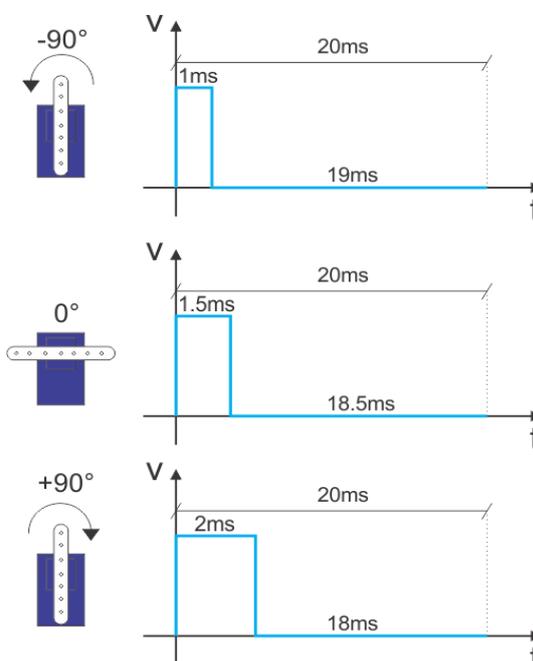
Fonte: <https://encrypted->

[tbn0.gstatic.com/images?q=tbn:ANd9GcTQCZFUay2WhREAs8YTs3h1YGgzF0M](https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTQCZFUay2WhREAs8YTs3h1YGgzF0M)

Para controlar os servos motores são utilizados o sinal PWM (*Pulse Width Modulation*), ou seja, a posição angular irá depender da largura de pulso enviado. Esta informação é codificada em modulação PWM através da largura do pulso em nível alto em relação ao período total de oscilação, ou seja, através do seu *duty cycle*. (EPUSP, 2014)

Em um servo motor, a posição é controlada com um sinal com período de 50Hz e pulsos com largura de 1 até 2ms, como mostrado na Figura 2.26.

Figura 2. 26 - Sinais de controle do servo motor.



Fonte: [https://2.bp.blogspot.com/sinal\\_controle.png](https://2.bp.blogspot.com/sinal_controle.png)

### 3 DESENVOLVIMENTO

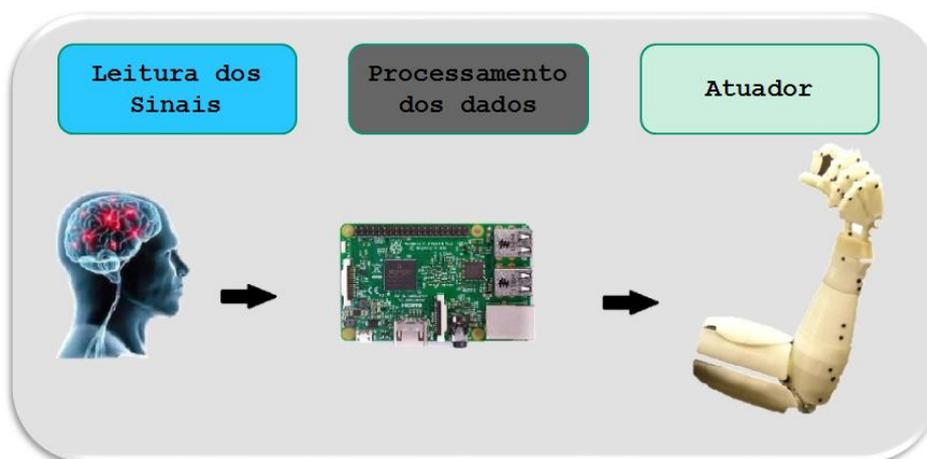
Neste capítulo serão demonstrados os métodos para o desenvolvimento deste projeto, assim como também a implantação e os testes baseados na metodologia deste projeto.

#### 3.1 Metodologia do projeto

Os métodos utilizados tiveram como foco a coletar dos dados emitidos pelo cérebro, aplicação dos dados em uma rede neural para tratá-los e atuação na saída, sendo o movimento pensado pelo indivíduo no início do processo.

O projeto pode ser dividido em três grupos principais, como mostra a Figura 3.1:

Figura 3. 1 – Divisão do projeto.



Fonte: Autor.

Onde, leitura dos sinais representa os sinais enviados pelo cérebro; processamento dos dados seria o processamento dos sinais através da rede neural artificial; e por último o atuador que é retratado pela prótese.

### 3.1.1 Leitura dos dados

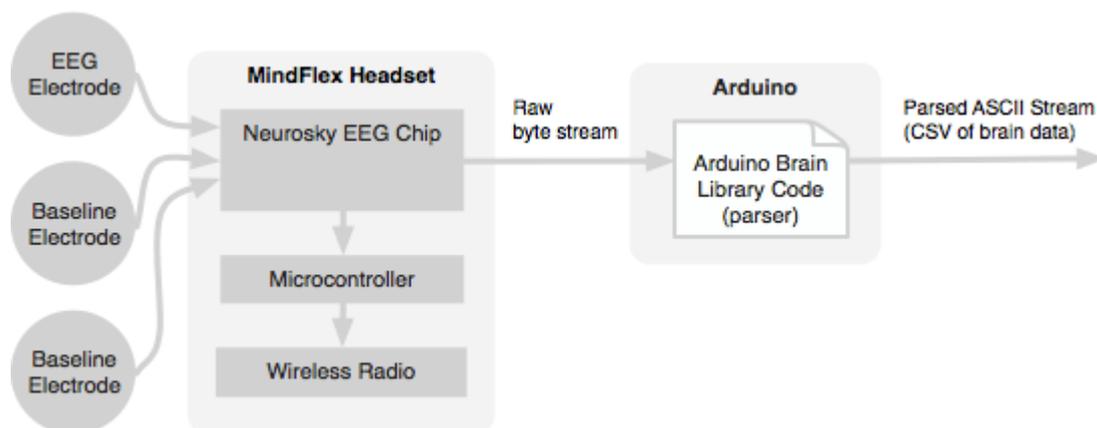
A fonte principal deste projeto é o cérebro humano, pois é nele que será coletado os sinais de entrada da RNA. Como o cérebro é o centro de controle do movimento, do sono, da fome, da sede e de quase todas as atividades vitais necessárias à sobrevivência, por isso foi utilizado como emissor dos sinais.

Estas ações foram geradas a partir de sinais elétricos emitidos pelo cérebro, na qual recebe o nome de ondas cerebrais. Que são as atividades rítmicas que o cérebro realiza o tempo todo, elas podem variar de acordo com o estado físico ou mental do indivíduo. O cérebro possui cinco tipos de ondas cerebrais, são elas: Gamma, Beta, Alpha, Theta e Delta.

Para realizar a leitura destes sinais foi preciso de um equipamento que não necessite de cirurgia para a colocação de eletrodos em contato com o cérebro, e sim algo que possa ser colocado e retirado facilmente do indivíduo. Para isso, foi utilizado os capacetes para leitura de cérebro.

O capacete *MindFlex* é um capacete capaz de efetuar a leitura das ondas cerebrais. Utilizando a plataforma Arduino para coletar estes dados, através de um código de programação conforme “Anexo A”. A coleta dos dados foram realizada direto do sensor EEG do capacete *MindFlex*, como mostra a Figura 3.2.

Figura 3. 2 - Modo de coleta dos sinais com o capacete.



Fonte: <http://frontiernerds.com/files/mindflex-schematic-hacked.gif>.

Assim os dados coletados são: qualidade do sinal, atenção, meditação, Delta, Theta, Alpha baixo, Alpha alto, Beta baixo, Beta alto, Gamma alto e Gamma baixo do indivíduo que está com o capacete. No quadro 3.1, nota-se os dados lidos pelo arduino e transmitidos para o computador.

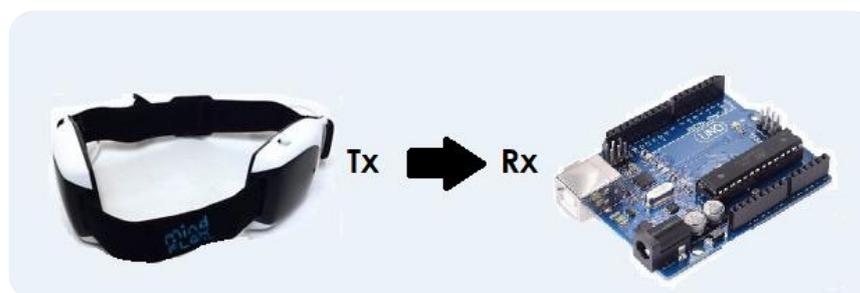
Quadro 3. 1 – Dados coletados pelo capacete.

Qualid.	Aten.	Meditação	Delta	Theta	Alpha Baixo	Alpha Alto	Beta Baixo	Beta Alto	Gamma Baixo	Gamma Alto
26	34	37	1242859	133512	110870	49896	118570	200011	143509	87778
55	34	37	64614	25869	50124	48316	34459	40090	33985	19915
26	0	0	707719	52529	29059	20200	31877	22495	15810	9204

Fonte: Autor.

Na Figura 3.3 é possível encontrar o fluxo de dados para a coleta dos sinais, onde o capacete realiza a leitura e o Arduino recebe as informações e transmite via comunicação serial para o Raspberry.

Figura 3. 3 - Fluxo de dados para coleta de informações.



Fonte: Autor.

Para referenciar quando se inicia o movimento com o sinal emitido pelo cérebro foi utilizado o sensor *flex*, que pode mudar a sua resistência conforme dobrada (o *datasheet* do sensor *flex* está disponível no “Anexo B”). Colocados em uma luva para acompanhar o movimento dos dedos do indivíduo, assim mostrando o momento onde se inicia o movimento do membro.

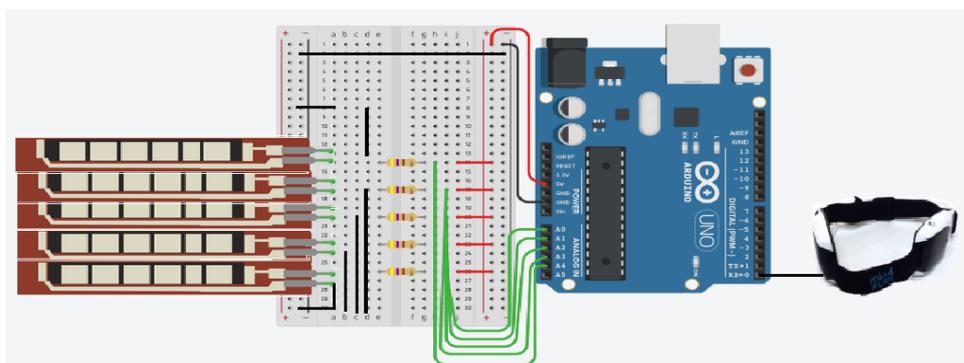
Figura 3. 4 – *Hardware* de coleta.



Fonte: Autor.

A Figura 3.4 mostra o *hardware* para a coleta dos sinais cerebrais, com o capacete, luva com os sensores e o Arduino. Já a Figura 3.5, mostra a ligação elétrica entre o *hardware* de coleta do projeto

Figura 3. 5 – Esquema elétrico da ligação do capacete com o Arduino.



Fonte: Autor.

O quadro 3.2 mostra os sinais do sensor *flex* com o cérebro. Se o indivíduo pensar em um movimento, deverá reproduzir o mesmo movimento não mão em que estão os sensores, assim os mesmos vão ser flexionados e apresentará uns valores que será de referência para a análise dos sinais do cérebro, mostrando onde começa e termina o movimento realizado.

Quadro 3. 2 - Dados com sensor *flex*.

Qualid.	Aten.	Meditação	Delta	Theta	Alpha		Beta		Gamma		Polegar	Indicador	Médio	Anular	Mínimo
					Baixo	Alto	Baixo	Alto	Baixo	Alto					
26	0	0	2500615	191598	140154	96280	80519	201084	118962	37125	165	154	156	149	146
55	0	0	1770248	230645	16250	36621	80909	112643	96244	80278	167	154	157	148	146
26	0	0	634185	199072	57628	86571	72571	169738	73606	32158	166	153	155	149	146
26	0	0	152328	26505	13206	52044	34275	52832	15076	13597	167	154	157	148	146
55	0	0	1700386	160597	122581	177452	110451	123650	89141	30585	167	154	155	149	147
26	0	0	327909	37000	10327	37841	25791	11428	19476	45261	166	155	155	148	146
55	0	0	303802	25103	27805	46019	50254	22710	13922	14158	166	155	155	147	146
26	0	0	219008	387814	91172	298394	65029	168143	95414	143378	168	154	156	148	146
26	0	0	1480811	346285	64975	78751	89582	133638	94063	39561	170	152	155	148	148
26	0	0	859393	721514	69885	113716	31591	88740	148303	50559	169	154	156	148	147
26	0	0	348026	1320767	4790	171225	53569	170796	43486	35329	167	154	154	148	149
55	0	0	661059	18280	8374	26490	12112	17432	17679	9727	166	154	155	148	146
26	0	0	1347880	96048	21354	79202	139124	78504	60125	44844	168	154	155	148	147
55	0	0	1304305	17568	19534	32504	10722	21923	11677	13949	167	154	156	147	148
26	0	0	1540531	86324	11187	51487	173760	126355	70755	113940	164	154	157	149	146
26	0	0	393836	341520	31711	99694	80962	112958	70740	80475	166	154	155	148	146
55	0	0	2051694	370846	241763	112520	50492	136045	116425	77806	166	154	155	148	147

### 3.1.2 Processamento dos sinais

Com os sinais podem ser coletados, o projeto entrou na fase processamentos destes dados, para isso foi utilizado como placa para o processamento o Raspberry, onde possui capacidade e velocidade elevada de processamento, justamente o que foi preciso para fazer um dispositivo que consiga ler o cérebro, processar e executar uma ação no final em um curto intervalo de tempo.

Os sinais são coletados pelo Raspberry, como mostrado no código do “Anexo C”. Os dados recebidos pelo processador são do tipo *string*, o código recebe e transforma-os em dados do tipo inteiro, assim podendo ser manipulados individualmente, já que foi preciso ajustar os mesmo para serem inseridos pela rede neural. Assim, os dados são transformados em arquivos txt, para serem utilizados pela RNA.

Com os movimentos definidos, no caso abrir e fechar a mão, se torna possível a criação do *dataset*. Onde estará concentrado todos os dados que representa estes movimentos, para o desenvolvimento deste projeto foram utilizados cerca de 2970 linhas de dados (cada linha possui os 11 sinais coletados pelo cérebro) para a criação do *dataset*. O valor total de dados foi distribuído de forma igualitária entre os três possíveis sinais de saída da rede, de forma que cada um possui-o um conjunto de dados com 990 linhas de dados do cérebro. Estes dados foram classificados como ‘2’

para significar o estado “neutro” do cérebro (para retratar outro estado do cérebro sem ser realizando os movimentos definidos para este projeto), ‘1’ para representar o momento em que a mão do indivíduo estava fechada, e ‘0’ para representar a mão aberta. Como mostrado no quadro 3.3, onde segunda linhas de dados representam os sinais enviados pelo capacete no momento em que o indivíduo estava com a mão fechada e pensando nesta ação, na terceira representa o mesmo caso porem com a mão aberta e na última linha representa um sinal neutro.

Quadro 3. 3 - Classificação dos dados para armazenamento no *dataset*.

Q.	Aten.	Meditação	Delta	Theta	Alpha Baixo	Alpha Alto	Beta Baixo	Beta Alto	Gamma Baixo	Gamma Alto	Sinal
26	34	37	1242859	133512	110870	49896	118570	200011	143509	87778	0
55	34	37	64614	25869	50124	48316	34459	40090	33985	19915	1
26	0	0	707719	52529	29059	20200	31877	22495	15810	9204	2

Fonte: Autor.

Neste momento, foram tirados os sensores *flex* como referência, pois o projeto não terá uma realimentação do sistema. Ficará dependente apenas dos sinais coletados do cérebro do indivíduo, já que a rede possui como referência os dados inseridos em seu *dataset* (conjunto de dados).

Os sinais que serão de entrada para a RNA estarão normalizados, ou seja, cada dado será dividido pelo maior valor que se encontra no *dataset*. No quadro 3.4 mostra quais os valores usados neste projeto.

Quadro 3. 4 – Valor de cada sinal para normalizar os sinais de entrada da RNA.

Q.	Aten.	Med.	Delta	Theta	Alpha Baixo	Alpha Alto	Beta Baixo	Beta Alto	Gamma Baixo	Gamma Alto
80	100	100	35838995	3197480	1942363	1779008	1354265	1657055	1639685	959320

Fonte: Autor.

A RNA pode ser dividida da seguinte forma: primeiramente, ocorre um processo de aprendizado dos dados, introduzidos de forma que a RNA “aprenda” a classificar cada sinal para ser executado o movimento. Depois do sistema treinado, é possível

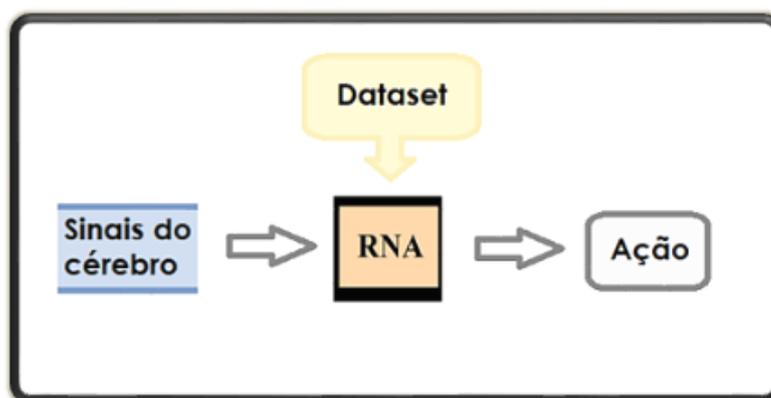
realizar a leitura de novos dados para serem classificados em determinado movimento. Como mostrado no código no “Anexo D”.

O modelo de rede *MLPClassifier* é utilizada para a realização de classificação de sinais e conforme venha a apresentar erro em sua saída realiza a correção dos seus pesos interno, no quais influenciam em seus neurônios. No caso do projeto são utilizados cerca de 28 neurônios em sua camada interna para a obtenção do resultado desejado.

O *dataset* pertence ao processo de treinamento da rede, onde ao iniciar o programa a rede sempre será treinada com estes valores. No caso, 70% dos dados do *dataset* são utilizados para realizar a classificação e 30% para validar os testes elaborados para classificar o valor de entrada.

A saída da rede representa qual a classificação que foi gerada a partir de um sinal de entrada, onde será comparado e executará o programa de ativação do braço robótico, podendo ser mão aberta, mão fechada e manter o movimento anterior ao que está sendo processado. A Figura 3.6 mostra a estrutura básica da rede no projeto, identificando os pontos de entrada, saída e o conjunto de dados para treinamento da rede.

Figura 3. 6 - Estrutura da Rede Neural.



Fonte: Autor.

### 3.1.3 Atuação no braço robótico

A saída da RNA será o sinal que representa o movimento que o indivíduo estará pensando, este sinal será a principal fonte para o acionamento do atuador deste

projeto, que no caso é a prótese.

As próteses são dispositivos artificiais que possuem a função de substituir parte do corpo de um indivíduo que por algum motivo perdeu ou nasceu sem um dos membros. O modelo de base para o desenvolvimento do projeto foi o robô InMoov, criação do francês Gaël Langevin, que se trata de um robô confeccionado a partir de impressão 3D.

A impressão 3D possui uma tecnologia de adição de camadas sobrepostas de materiais. Seu material utilizado é o filamento PLA ou Ácido Polilático, derivado a partir do milho e outros amidos renováveis; filamento de fácil impressão, indicado para peças que serão expostas a esforços de abrasão.

A prótese 3D possui 5 servos motores que são responsáveis por realizar a movimentação de cada dedo da mão artificial. O modelo empregado foi o servo motor – MG995 como mostrado na Figura 3.7, com dimensões de aproximadamente 40.7 x 19.7 x 42.9 mm, ideal para o encaixe na base do braço da prótese. Sua tensão de operação é de 4.8V a 7.2V, com corrente de 500mA (como mostra seu *datasheet* que se encontra no “Anexo E”).

Figura 3. 7 – Servo Motor.



Fonte: [http://aerocg.com.br/wp-content/uploads/2016/08/84368\\_1\\_High.jpg](http://aerocg.com.br/wp-content/uploads/2016/08/84368_1_High.jpg).

O sistema que é ligado em cada dedo ao motor se dá através de uma linha de algodão, se o motor girar no sentido horário o dedo da prótese começa a se abrir, se o motor girar no outro sentido o dedo começa a fechar. A escolha do material de ligação dos dedos com os motores se deu a partir de testes realizados com náilon, fio de cobre, barbante e linha de algodão, que no caso apresentou melhor resultado em movimentar os dedos.

Com a prótese já montada, foi possível testar a movimentação dos dedos de forma conjunta. Utilizando o código criado do “Anexo F” para a manipulação dos motores e ajustes dos valores que correspondem com os ângulos que os atuadores estão em função da posição dos dedos da prótese.

### 3.2 Implementação e testes

A primeira etapa para a implantação do projeto foi a coleta dos dados e a busca para encontrar padrões existentes nos movimentos predefinidos. Os dados coletados para o desenvolvimento do projeto, foram os da própria aluna Tamires, bem como os dados composto pelo *dataset*.

Para deixar o projeto mais otimizado foi implementado a coleta dos sinais do cérebro através do modulo *bluetooth*. O módulo ficava fixado ao capacete que se conectava com o Raspberry, para poder transmitir os sinais recebidos.

Mas ao realizar os testes em um ambiente com aparelhos eletrodomésticos, foi possível notar que os dados recebidos não representavam nenhuma informação legível, como visto no quadro 3.5, pois se tratava de valores de grandeza pequena (como o número 0, por exemplo). Os sinais de ‘qualidade do sinal’ variavam entre 25 e 26, ‘atenção’ se permanecia constante em 29 e ‘meditação’ também apresentava um número constante no valor de 75.

Quadro 3. 5 – Dados recebidos pelo módulo *Bluetooth*.

Delta	Theta	Alpha baixo	Alpha alto	Beta baixo	Beta alto	Gamma baixo	Gamma alto
1572864	393216	6488064	720896	3473408	4390913	7208963	5832704
1575287	7078401	4784219	8978605	7930026	13303855	14	8650757
1572864	0	0	0	0	0	0	0
1572864	131072	0	0	65536	131072	0	0
1572864	0	196608	0	0	65536	0	0
1572864	0	0	65536	65536	0	0	0
1572864	131072	0	0	0	0	0	0
1572864	589824	0	65536	0	0	720896	983040
1572864	65536	262144	0	131072	196608	2883584	2949120

Fonte: Autor.

Para que não houvesse alteração na interpretação dos dados do cérebro, optou-se por fazer a ligação de forma direta, utilizando um Arduino como interface de comunicação entre o capacete e o Raspberry. Já que a tensão do pino RX (pino receptor) do Raspberry é diferente da tensão do pino TX (pino transmissor) do capacete, por este motivo foi empregado a porta serial USB do Arduino.

Com a comunicação estabelecida, foi iniciada a fase de coleta e análise dos dados recebidos. Para que os testes possuíssem um padrão de repetição, foi estabelecido alguns movimentos iniciais para poder ser comparado com o resultado.

Os primeiros testes realizados foram com o movimento de abri e fechar o braço, a posição do membro estava perpendicular ao corpo, o movimento seguia de frente do corpo para a lateral e vice-versa.

Apesar do sistema estar mais estável, ainda sim com este perfil de movimento se tornava possível interpretar e analisar os sinais cerebrais do movimento de forma equivocada.

Com o intuito de absorver o máximo de informações do movimento optou-se pela troca do mesmo. Assim, os testes passaram a ser realizados com base nos movimentos da mão, pois se tratava de um movimento menor comparado com o do braço e tornando possível a implantação no projeto. Assim colocando uma luva com sensores *flex*, para obter uma referência entre o momento em que o indivíduo pensa no movimento e o modo de resposta do membro.

Foram realizados testes com a utilização de uma bolinha e um copo, porém os padrões encontrados ainda não eram suficientes para comprovar o movimento que o indivíduo estava pensando. Assim optou - se por apenas dois movimentos iniciais já que estes dados de base serviriam como classificador para a rede, os movimentos foram o de abrir e fechar a mão.

Com os movimentos definidos foi iniciado a elaboração do *dataset*, onde consta todos os dados de referências deste projeto. Para sua criação foi disponibilizado três tipos de sinais, os que representam o movimento de mão fechada (recebem o valor 0), o de movimento de mão aberta (recebem o valor 1) e o neutro (recebem o valor 2) que retrata o cérebro em estado diferente em relação aos movimentos, para demonstrar outras atividades não mapeadas que o indivíduo esteja realizando quando estiver com o dispositivo conectado.

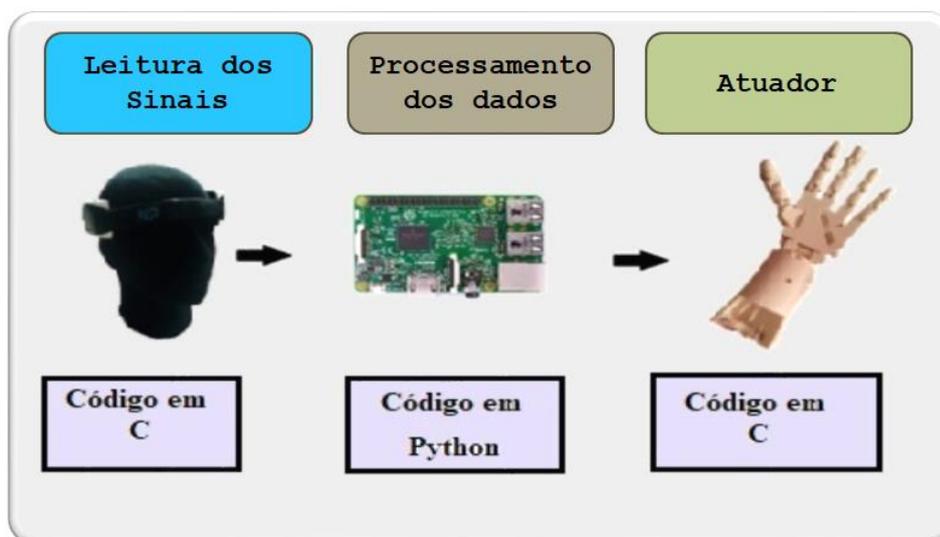
Na gravação dos dados do *dataset*, o indivíduo apenas pensava no movimento em, pensando nas palavras ‘mão aberta’ e ‘mão fechada’, esquecendo do que estava acontecendo ao seu redor. No caso do neutro, nada foi pensado no momento do desenvolvimento, já que este seria o sinal que representa o oposto dos dados dos movimentos determinados.

### 3.2.1 Junção das etapas do projeto

Como foi dito no início deste capítulo, o projeto está dividido em três partes principais leitura dos sinais que representa os sinais enviados pelo cérebro, processamento dos dados que corresponde ao processamento dos sinais através da RNA, e por último o atuador que foi representado pela prótese.

Assim como o projeto foi dividido em três partes para um melhor entendimento de seu funcionamento, a programação também foi subdividida em três partes. Onde cada parte foi implementada de forma individual, como mostra a Figura 3.8.

Figura 3. 8 – Divisão do projeto junto pela programação.



Fonte: Autor.

Como analisado na figura acima, cada parte do projeto está em uma linguagem de programação diferente da sua anterior. Isso ocorreu para que de forma individual

cada parte do projeto pudesse ser testada e ajustada de acordo com a aplicação. Tendo o receio de que fosse implementado em sequência, tornaria difícil encontrar qualquer erro que surgisse durante a implantação do código.

Mesmo que os códigos funcionem de forma individual, ainda assim optou-se por dividir a junção em três partes para obter o resultado desejado deste projeto.

### 3.2.1.1 Leitura e processamento

Os códigos de cada parte do projeto se encontram em uma linguagem diferente de programação. No caso da leitura está em linguagem C e o processamento em Python.

Ao coletar os dados emitidos pelo cérebro o programa faz a gravação em dois arquivos de texto, o primeiro com os valores puros do cérebro (se caso seja necessária uma análise destes sinais, assim não havendo perdas do mesmo) e o segundo com os valores coletados normalizados pelos valores máximos encontrados em cada sinal quando desenvolvido o *dataset* deste projeto.

Quando o processamento, a rede, recebe um dado de entrada através do arquivo de texto, antes de realizar a classificação os dados são normalizados, já que no *dataset* da rede os valores estão todos normalizados.

Como a coleta dos sinais ocorre de forma instantânea, a cada segundo um novo sinal é transcrito no arquivo criado; a rede realiza apenas a leitura da última linha do arquivo dos sinais normalizados. A Figura 3.9 mostra como esta leitura do arquivo normalizado foi realizada no programa da rede neural.

Figura 3. 9 – Leitura do arquivo normalizado.

```
1 tmp = os.popen("tail -n 1 cerebro1.txt | tr -d '\n\' | tr '\;\' \' \'")
2
3 cerebro = tmp.readline()
4 tmp.close()
5
```

Fonte: Autor.

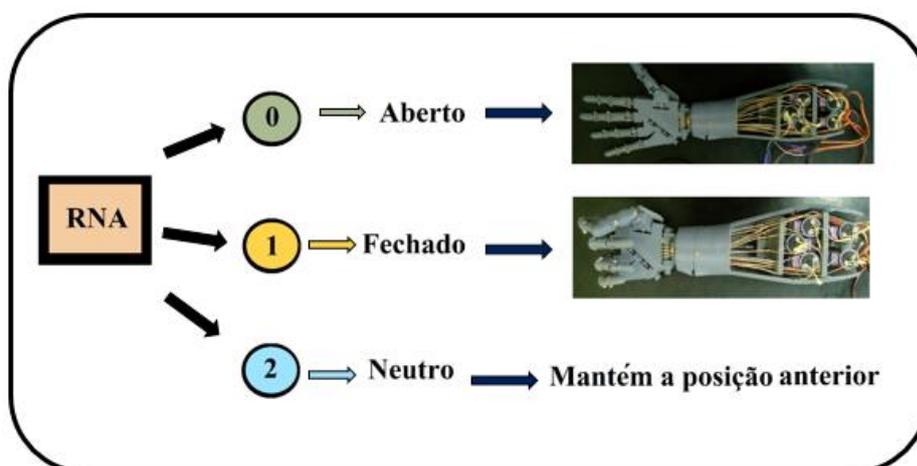
A variável 'cérebro' recebe o valor dos dados do cérebro do arquivo normalizado, assim podendo ser classificado pela rede neural.

### 3.2.1.2 Processamento e atuador

No caso do processamento com o atuador seria necessário trocar informações entre Python e Linguagem C, pois a rede tem a função de mandar um sinal para o programa de controle da prótese para atuar nos servos motores. Para facilitar esta troca de informação, no programa da rede foi utilizado o comando '`os.system ("file name")`', para chamar o arquivo executável de manipulação dos motores. Assim quando a rede classificar o valor recebido, o seu resultado será comparado para determinar qual arquivo de movimento será realizado pela prótese.

Se o resultado da rede for '0', será executado o arquivo de abrir a mão, mas se a prótese estiver aberta os motores continuaram parados. Mas se o resultado for '1', o arquivo será o de fechar a mão; caso o resultado seja '2' nada ocorrerá com os motores, pois este seria um "estado neutro" do cérebro onde a rede estará treinada para não enviar nenhum sinal para os programas de atuação do projeto; como mostra a Figura 3.10.

Figura 3. 10 - Saída da Rede Neural.



Fonte: Autor.

Como o projeto não possui uma realimentação do sistema de forma física, optou - se por colocar uma espécie de realimentação pela programação, onde se o sistema já tenha detectado um movimento, não precisasse refazer este mesmo movimento já que a prótese estaria no local desejado. Esta realimentação pela programação pode ser vista através da Figura 3.11, que mostra o código em python do trecho referente.

Figura 3. 11 – Realimentação do sistema.

```

1  if valor == '0':
2      if(t==1):
3          print("Mão já está aberta")
4      else:
5          t=1
6          print("Mão aberta")
7      #Arquivo de abrir a mão
8          os.system("/media/tamires/Dados1/TCC/PWM/pwm_MA")
9
10 if valor == '1':
11     if(t==2):
12         print("Mão já está fechada")
13     else:
14         t=2
15         print("Mão fechada")
16     #Arquivo de fechar a mão
17         os.system("/media/tamires/Dados1/TCC/PWM/pwm_MF")
18
19 if valor == '2':
20     print("Neutro")

```

Fonte: Autor.

Vale ressaltar que os programas para manipulação dos motores, tem a função de transferir de um estado para o outro, como por exemplo, se a prótese está aberta e a rede recebe um sinal e em sua saída o resultado for igual a '1', o arquivo para 'mão fechada' será acionado, assim passando do estado de mão aberta para mão fechada; da mesma maneira como a mão humana realiza na necessidade de abrir e fechar o membro.

### 3.2.1.3 Leitura, processamento e atuação

Com os programas já estabilizados para conversarem entre eles, foi realizado a implementação dos três códigos de forma simultâneas. Primeiro rodando o arquivo para coleta dos dados do cérebro, em seguida o arquivo da rede neural e os dos motores da prótese que são chamados pelo resultado da rede.

A Figura 3.12 demonstra o resultado da compilação dos códigos na tela do Raspberry.

Figura 3. 12 – Monitoramento dos programas.

The screenshot shows a Raspberry Pi desktop with several terminal windows. The top-left window displays a list of data points. The top-right window shows system performance metrics like CPU usage, memory, and a process list. The bottom-left window shows the execution of a Python script with a traceback error. The bottom-right window shows the output of the same script, displaying numerical results for each iteration.

```

root@tamires-desktop: /media/tamires/Dados1/TCC
File Edit View Search Terminal Help
80,0,0,1089604,199952,66366,35200,36747,284870,266025,45037,58,56,59,60,64
51,0,0,1001102,196103,12540,42466,74782,248321,397412,76724,10,10,11,12,17
51,0,0,173650,15885,5501,5738,1249,53119,71792,22408,0,0,0,0,0
80,0,0,41972,79382,9936,14429,11913,143145,28564,24002,58,60,60,61,65
51,0,0,95075,28107,8452,16396,2623,164287,28468,13408,2,5,4,7,16
51,0,0,533020,41070,5027,13802,45287,146467,236471,39358,0,0,0,0,0
^C
root@tamires-desktop: /media/tamires/Dados1/TCC#

tamires@tamires-desktop: ~
File Edit View Search Terminal Help
1  [ ]  2.6%  Tasks: 117, 147 thr; 2 running
2  [ ]  3.2%  Load average: 0.42 0.45 0.81
3  [ ]  1.3%  Uptime: 01:47:21
4  [ ]  8.3%
Mem [|||||] [328M/926M]
SwP [|||||] 0K/0K

PID USER      PRI  NI  VIRT   RES   SHR  S  CPU%  MEM%  TIME+  Conn
1157 tamires    20   0  78668 45896 21452  S   2.6  4.8  2:08.14 /usr
1329 tamires    20   0  63144 17544 14820  S   2.6  1.9  0:04.52 /usr
2815 tamires    20   0  5220  2384  2008  R   1.9  0.3  0:57.64 /htop
980  xrdp        20   0  22252 10504 1900   S   1.3  1.1  1:35.68 /usr
1137 xrdp        20   0  22252 10504 1900   S   1.3  1.1  1:35.67 /usr
F1 help F2 setup F3 search F4 file F5 free F6 sort by F7 nice F8 nice F9

root@tamires-desktop: /media/tamires/Dados1/TCC/RNA/comparacao
File Edit View Search Terminal Help
-rw-r--r-- 1 root root 2048582 Mai 6 11:30 dados_20180508_
-rw-r--r-- 1 root root 2421668 Mai 6 11:45 dados_20180508_
drwxrwxr-x 2 tamires tamires 4096 Abr 30 22:46 dados_antigos
-rw-rw-r-- 1 tamires tamires 320024 Abr 30 22:54 dados_entrada.c
-rw-rw-r-- 1 tamires tamires 8910 Abr 30 22:43 dados_saida.csv
-rw-r--r-- 1 root root 27 Mai 1 14:39 dados_teste.csv
-rw-r--r-- 1 root root 141 Mai 1 14:41 teste.py
root@tamires-desktop: /media/tamires/Dados1/TCC/RNA/comparacao# py
thon compV4.py >> dados_20180508_1446.txt
/usr/local/lib/python2.7/dist-packages/sklearn/neural_network/mul
tilayer_perceptron.py:912: DataConversionWarning: A column-vector
y was passed when a 1d array was expected. Please change the sha
pe of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
^CTraceback (most recent call last):
  File "compV4.py", line 43, in <module>
    cerebro = tmp.readline()
KeyboardInterrupt
root@tamires-desktop: /media/tamires/Dados1/TCC/RNA/comparacao#

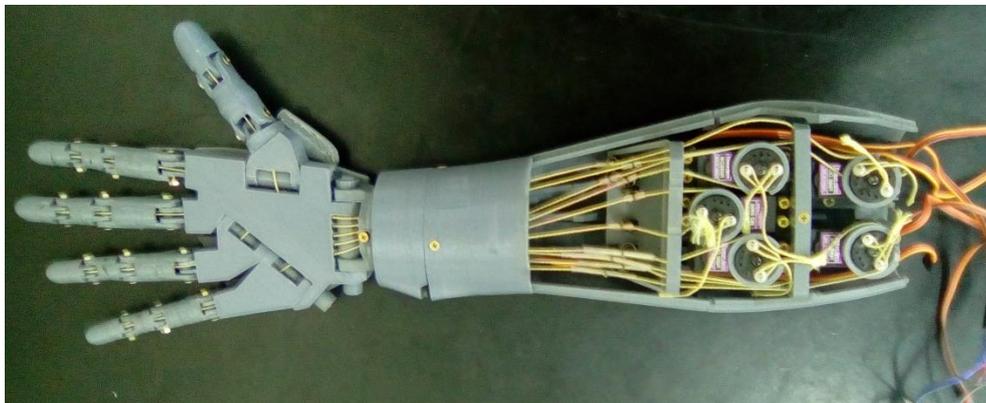
"tela.png" selected (322.5 kB), Free space: 7.2 GB
root@tamires-deskto...  Dados1  root@tamires-deskto...  tamires@tamires-desk...  root@tamires-desktop...

```

Fonte: Autor.

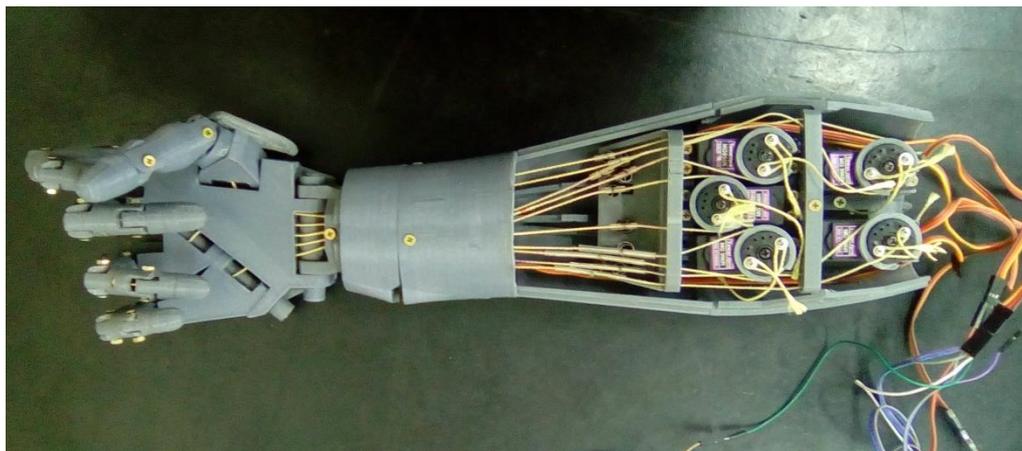
Já as Figuras 3.13 e 3.14 representam como a prótese ficará com o movimento de mão aberta e fechada respectivamente.

Figura 3. 13 – Movimento de mão aberta.



Fonte: Autor.

Figura 3. 14 – Movimento de mão fechada.



Fonte: Autor.

No momento de classificação da rede para o sinal neutro, a prótese não realiza nenhum movimento, permanece na mesma posição anterior, ou seja, se a mão estava aberta ficará aberta até que o próximo sinal do cérebro indique que o indivíduo trocou o movimento.

A Figura 3.15 apresenta as ligações físicas do projeto ao final do desenvolvimento de cada módulo.

Figura 3. 15 - Estrutura física do projeto.



Fonte: Autor.

## 4. RESULTADOS E DISCUSSÕES

Com a aplicação e o desenvolvimento do hardware e do software do projeto, foi possível obter alguns resultados que serão apresentados neste capítulo.

### 4.1 Primeiro módulo do projeto

A implementação do primeiro módulo do projeto, se dá a partir da leitura dos dados do cérebro do indivíduo. Com esta comunicação estabelecida, é possível desenvolver o *dataset* do projeto, com os dados dos movimentos que poderão ser reproduzidos pelo braço robótico. O quadro 4.1 mostra alguns dados existentes no *dataset* do projeto, vinculando o dado do movimento com a sua numeração de saída, ou seja, o número ao qual o grupo de movimento que o sinal pertence.

Quadro 4. 1 - Dados do *dataset*.

Qualidade	Atencao	Meditacao	Delta	Teta	Alfa_baixo	Alfa_alto	Beta_baixo	Beta_alto	Gama_baixo	Gama_alto	Saída
32,5	34	37	9,814963887	8,545823586	7,769917364	1,394597439	10,60656518	4,376801011	11,13652927	4,841137472	0
32,5	34	37	9,814963887	8,545823586	7,769917364	1,394597439	10,60656518	4,376801011	11,13652927	4,841137472	0
68,75	34	37	13,95180383	2,954295258	1,380534946	9,903328147	4,246436259	7,301568143	6,155023678	10,1201893	0
32,5	34	37	25,34931966	7,958079488	2,206024312	3,774856549	7,386146729	5,738252502	7,201017269	8,047679606	0
32,5	34	37	18,52732851	7,459092786	8,601378836	6,214699428	6,72977593	13,51874259	10,91343764	16,56923654	0
68,75	34	37	5,323370244	5,806791598	2,240621346	3,776655305	9,404880138	19,24854637	11,965347	10,06431639	0
32,5	34	37	25,75396322	20,18192452	7,509461414	4,600485214	4,019818869	5,567708978	16,66917731	4,180044198	0
32,5	34	37	29,11544563	21,72560892	28,66729854	29,50200336	59,14647429	77,91835515	27,00036897	25,68402619	0
68,75	34	37	32,0864032	6,304621139	4,573347001	3,093971472	11,19633159	12,5595107	11,87337812	14,53738064	0
68,75	34	37	19,11604553	8,111481542	1,673013747	2,02567948	4,080959044	7,733418625	11,21038492	9,336404954	1
68,75	34	37	8,571986623	10,91049827	12,76954926	5,667709195	7,601171115	11,88524219	9,827070443	8,403452445	1
32,5	34	37	29,87244325	17,90860302	19,02733938	33,85892587	30,2371397	65,19759453	70,13652012	47,42088146	1
68,75	34	37	0,674657042	0,883727185	1,346040879	1,263232093	1,256253392	1,292171956	0,981164065	1,450506609	1
68,75	34	37	1,802898801	0,809043372	2,580568102	2,7158956	2,544479847	2,419352405	2,072654199	2,075949631	1
32,5	34	37	7,355237807	10,44325531	15,4771276	6,684680451	9,752670268	4,813901772	6,699701467	10,58187049	1
32,5	34	37	4,880890763	2,117667663	2,45052032	1,101512753	2,144484277	1,531934667	0,901026722	1,445503065	1
68,75	34	37	17,43636463	4,98398739	30,23770531	6,50148847	4,922817912	6,326404374	2,446079582	9,029625151	1
32,5	34	37	1,292336969	3,948484431	1,206005263	0,175266216	0,915256615	1,301043116	1,900730933	2,074386023	1
32,5	34	37	25,3690189	11,77958893	20,70483221	9,537112818	18,95891868	6,038483937	7,198333826	9,938289622	1
68,75	0	0	8,476866649	0,785087006	1,431503792	2,586778699	3,710795155	1,370503695	0,849065522	1,475837051	2
32,5	0	0	6,110893316	12,12873888	4,693870301	16,77305555	4,801792854	10,14709832	5,819044512	14,94579494	2
32,5	0	0	41,31848171	10,82993482	3,345152271	4,426680487	6,614805817	8,064789642	5,736650637	4,123858566	2
32,5	0	0	23,97930185	22,5650825	3,597937152	6,392101666	2,332704456	5,355283922	9,044603079	5,270295626	2
32,5	0	0	9,710831372	41,30649762	0,246606839	9,624745926	3,95557738	10,30720163	2,652094762	3,682712755	2
68,75	0	0	18,44526695	0,571700214	0,431124357	1,48903209	0,894359671	1,051986808	1,078194897	1,013947379	2
32,5	0	0	37,60936077	3,003865544	1,099382556	4,452031694	10,27302633	4,737561517	3,666862843	4,674561147	2
68,75	0	0	36,39350483	0,549432678	1,005682254	1,827085657	0,79172097	1,323009798	0,71214898	1,454050786	2
32,5	0	0	42,98482517	2,699751054	0,575947956	2,894141004	12,830576	7,625274961	4,315158094	11,87716299	2
32,5	0	0	10,98904962	10,68091122	1,632599056	5,603909595	5,978298191	6,816792442	4,314243285	8,388754534	2

Fonte: Autor.

Com o objetivo de filtrar os sinais existentes no *dataset*, foi realizada uma seleção dos dados, onde foram descartados os sinais acima de 20 da categoria Delta,

Theta, Alpha baixo, Alpha alto, Beta baixo, Beta alto, Gamma baixo e Gamma alto, para o movimento de abrir e fechar a mão. Já a ação neutro do projeto, foram mantidos os sinais abaixo de 20 para as mesmas categorias.

Foi necessário realizar este filtro para auxiliar a rede neural em sua classificação, pois quando foram realizados a coletas dos dados para a elaboração do *dataset*, pode ter ocorrido uma perda de atenção ou de sinal na coleta dos movimentos do projeto.

## 4.2 Segundo módulo do projeto

Antes de iniciar a classificação de alguns sinais de entrada, a rede necessita ser treinada com os valores do seu *dataset*, como mostra a figura 4.1.

Figura 4. 1 - Treinamento da rede neural.

```
root@tamires-desktop:/media/tamires/Dados1/TCC/RNA/comparacao# python compV4.py
/usr/local/lib/python2.7/dist-packages/sklearn/neural_network/multilayer_perceptron.py:912: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Fonte: Autor.

Com a rede treinada, basta inserir um sinal de entrada que a mesma realiza a classificação de acordo com os dados inseridos em seu *dataset*, como mostra a Figura 4.2, onde a rede classifica os de entrada.

Figura 4. 2 - Classificação da rede neural.

```
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
Neutro
250.000000 0.000000 0.000000 13.111712 15.164817 1.057681 7.515761 6.820969 11.343800 7.528153 6.702039
['2']
```

Fonte: Autor.

Enquanto a rede realiza a classificação, é possível observar na tela os dados coletados em tempo real pelo capacete e também a normalização destes sinais quanto inseridos na rede neural. O quadro 4.2 mostra os dados coletados pelo capacete.

Quatro 4. 2 - Dados do capacete.

Qual.	Ate.	Med.	Delta	Theta	Alpha baixo	Alpha alto	Beta baixo	Beta alto	Gamma baixo	Gamma alto
26	0	0	650382	429796	41393	112332	91960	155767	149370	118980
55	0	0	709106	182544	12862	10143	6602	27126	36540	15326
26	0	0	1288215	603771	112157	59707	63655	222096	77264	135824
26	0	0	1375353	624987	181927	61766	167075	114652	110327	56331
55	0	0	224193	596523	165211	77810	91005	207632	65321	225183
26	0	0	345984	625531	543159	181774	65490	77219	116612	155697
55	0	0	588736	174408	9894	39561	120702	229205	160247	88210
51	0	0	661633	272251	49070	140976	233526	183465	65851	86242
26	0	0	552251	150851	98281	92046	66921	140868	108600	398028
80	0	0	633709	142777	612	98713	85514	108570	75730	139371
26	0	0	1358529	350214	233979	45195	97849	99372	150690	170962
51	0	0	178050	39291	41361	70322	38133	21288	19789	26890
55	0	0	568787	123423	229502	98264	258255	155852	59656	92939
26	0	0	97776	67443	101448	28057	17111	51338	26313	11543
55	0	0	501945	137733	93000	118750	270241	169959	218242	108088
26	0	0	1988800	1531726	445399	258496	907205	436907	1057087	136080
26	0	0	21793	47690	2523	27675	64885	48043	22074	36661
55	0	0	176395	90650	4695	21891	18009	26603	21151	18890
26	0	0	1964874	1580346	645103	1161698	596706	465303	558611	758842
26	0	0	1090672	733149	105454	88990	281927	92367	112669	137098

Fonte: Autor.

Já o quadro 4.3 mostra estes mesmo valores normalizados (ou seja, divididos pelo maior valor encontrado no conjunto de dados e multiplicado por 100, em busca de padronizar os valores do cérebro já que possuem grandezas diferentes em si) antes de serem inseridos na rede neural, que já estará treinada de acordo com os valores inseridos no *dataset*.

Quatro 4. 3 - Dados normalizados.

Qual.	Ate.	Med.	Delta	Theta	Alpha baixo	Alpha alto	Beta baixo	Beta alto	Gamma baixo	Gamma alto
32,5	0	0	18,14735	13,4417	2,131064	6,314305	6,79039	9,40023	9,109676	12,402534
68,75	0	0	19,785904	5,70899	0,662183	0,570149	0,487497	1,637001	2,228477	1,59759
32,5	0	0	35,944553	18,88273	5,774255	3,356196	4,700336	13,403055	4,712125	14,158361
32,5	0	0	38,375931	19,54623	9,366271	3,471935	12,33695	6,919022	6,728549	5,871972
68,75	0	0	6,255569	18,65603	8,505671	4,373786	6,719881	12,530181	3,983753	23,473188
32,5	0	0	9,653854	19,5632	27,963823	10,217717	4,835834	4,660014	7,111854	16,229933
68,75	0	0	16,427267	5,45454	0,50938	2,223767	8,912731	13,832068	9,773036	9,195055
63,75	0	0	18,461283	8,51454	2,526304	7,924416	17,24374	11,071751	4,016077	8,989909
32,5	0	0	15,409241	4,71780	5,059867	5,174007	4,9415	8,501105	6,623223	41,490639
100	0	0	17,682131	4,46529	0,031508	5,548767	6,314421	6,551985	4,61857	14,528103
32,5	0	0	37,906498	10,95281	12,0461	2,540461	7,225247	5,996904	9,19018	17,821165
63,75	0	0	4,968059	1,22881	2,129416	3,952877	2,815771	1,284689	1,206878	2,803027
68,75	0	0	15,870638	3,86000	11,815608	5,523528	19,06975	9,40536	3,63826	9,688008
32,5	0	0	2,728205	2,10925	5,222917	1,577115	1,26349	3,098147	1,604759	1,203248
68,75	0	0	14,005572	4,30754	4,787982	6,675068	19,954809	10,25669	13,309996	11,267147
32,5	0	0	55,492699	47,9041	22,93078	14,530345	66,988731	26,366474	64,46891	14,185047
32,5	0	0	0,608081	1,49148	0,129893	1,555642	4,79116	2,8993	1,346234	3,821561
68,75	0	0	4,92188	2,83504	0,241716	1,230517	1,329799	1,605438	1,289943	1,969103
32,5	0	0	54,8251	49,42473	33,212276	65,300323	44,061241	28,080116	34,068192	79,102074
32,5	0	0	30,432589	22,92896	5,42916	5,002226	20,817713	5,574166	6,871381	14,291164

Fonte: Autor.

Quando o sinal for classificado de acordo com os sinais do *dataset*, resultará em 3 possíveis saídas, como mostrado na Figura 4.4, sendo '0' ativação do programa de ativação do braço robótico para mão aberta, '1' ativação do programa de mão fechada e '2' para manter a posição anterior do braço robótico.

Figura 4. 3 – Vinculo do sinal de entrada com o de saída.

Qualidade	Atencao	Meditacao	Delta	Teta	Alfa_baixo	Alfa_alto	Beta_baixo	Beta_alto	Gama_baixo	Gama_alto	Sinal de saída
26	34	37	351758	273251	150920	24810	143641	72526	182604	46442	0
55	34	37	500018	94463	26815	176181	57508	120991	100923	97085	0
26	34	37	908493	254458	42849	67155	100028	95086	118074	77203	1
26	34	37	664000	238503	167070	110560	91139	224013	178946	158952	1
55	34	37	190784	185671	43521	67187	127367	318959	196194	96549	2
26	34	37	922995	645313	145861	81843	54439	92260	273322	40100	2



Fonte: Autor.

## 5. CONSIDERAÇÕES FINAIS

Neste capítulo, serão apresentadas as conclusões finais em relação aos resultados obtidos e sugestões para trabalhos futuros em relação ao projeto.

### 5.1 Conclusão

A implementação do projeto foi realizada em três módulos principais, leitura dos sinais, processamento dos sinais pela rede e movimentação da prótese.

A leitura dos sinais cerebrais foi realizada com o auxílio do capacete *MindFlex* onde foi possível analisar e interpretar o que os sinais estavam representando a partir de certos movimentos. Na coleta dos dados, obtive melhor resultado uma coleta via Arduino do que por via *bluetooth*.

Os movimentos implementados foram de fechar a mão e de abrir a mão. Porém, na elaboração da *dataset* (conjunto de dados dos movimentos existem neste projeto), onde foram inseridos os movimentos de mão aberta, fechada e neutro (para diferenciar dos demais sinais).

O sinal de neutro foi inserido para caso o cérebro pense em outro movimento que não seja mão aberta ou fechada, não executar os movimentos, respectivamente, na mão. Assim quando a rede entrar no processo de treinamento, ela chama estes dados para que possa compará-los com a sua entrada e devolver na saída qual o movimento corresponde o que o indivíduo pensou.

Por tanto, após a implementação de todas as etapas do projeto individuais e em conjunto, se torna possível o controle da prótese através da interpretação do cérebro de qualquer indivíduo, desde que haja a coleta de dados para a organização de um novo *dataset*.

### 5.2 Trabalhos futuros

Com base nos resultados obtidos as adaptações futuras em relação ao projeto são:

- Realimentação da prótese via *hardware*

Buscando saber onde exatamente se encontra a posição dos dedos, assim podendo auxiliar para a implementação de novos movimentos no projeto.

- Ajuste nos códigos para padronização das linguagens de programação

Tornar-se o projeto em apenas uma linguagem de programação, podendo ser transferidos os módulos para python ou 'C'.

- Aumentar a quantidade de movimentos

Como os resultados foram positivos para o movimento de abrir e fechar a mão, se torna possível implementar novos movimento, desde que haja a coleta dos novos dados para adaptação do *dataset* do sistema.

- Deixar o projeto mais otimizado

Melhorando a “aparência física” do projeto para que possa ser implementado em um indivíduo que venha ser amputado. Implementar uma bateria de longa duração e aumentar a resistência dos dispositivos utilizados.

- Diminuir o número de dispositivos

Com o objetivo de deixar o projeto funcional para um indivíduo utiliza-lo em seu dia a dia seria necessário diminuir o número de dispositivos no projeto.

- Ampliar o projeto para outros membros

Assim como é possível aumentar a quantidade de movimentos é possível também estender o projeto para outros membros, já que seria utilizado a mesma base de implementação (leitura dos dados para a realização de um novo *dataset*).

## 6. REFERÊNCIAS

A descoberta da bioeletricidade. Disponível em: <<http://www.cerebromente.org.br/n06/historia/bioelectr.htm#author>>. Acesso em 07 de setembro de 2017.

A eletrocorticografia como meio de orientação na terapêutica cirúrgica de epilepsia. Disponível em: < [http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S0004-282X1952000300012](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0004-282X1952000300012)>. Acesso em 20 de agosto de 2017.

Abrahams, Peter. Atlas descritivo do corpo humano. São Paulo: Rideel, 2009. 1e.

Bear, Mark F.; Connors, Barry W.; Paradiso, Michael A. Neurociências – Desvendando o sistema nervoso. São Paulo: Artmed, 2017, 4e.

CENSO 2010. Disponível em: <<https://censo2010.ibge.gov.br/noticias-censo.html?busca=1&id=3&idnoticia=2170&t=censo-2010-numero-catolicos-cai-aumenta-evangelicos-espíritas-sem-religiao&view=noticia>>. Acesso em 20 de agosto de 2017.

CONHEÇA os diferentes tipos de materiais para impressão 3D. Disponível em: <<http://www.impressao3dfacil.com.br/conheca-os-diferentes-tipos-de-materiais-para-impressao-3d-fdm/>>. Acesso em 10 de setembro de 2017.

CONTROLE de um servo motor – EPUSP. Disponível em: <[https://www2.pcs.usp.br/~labdig/pdf/files\\_2014/control-servo-semesteral.pdf](https://www2.pcs.usp.br/~labdig/pdf/files_2014/control-servo-semesteral.pdf)>. Acesso em 20 de abril de 2018.

CÓRTEX Cerebral – Regiões e Funções – O que é? Disponível em: <<http://www.anatomiadocorpo.com/sistema-nervoso/cerebro/cortex-cerebral/>>. Acesso em 27 de agosto de 2017.

DIFERENTES deficiências e seus conceitos. Disponível em: <<http://www.ppd.mppr.mp.br/modules/conteudo/conteudo.php?conteudo=17>>. Acesso em 20 de agosto de 2017.

EMPRESA privada e neurocientista brasileiro criam robôs diferentes que permitem que paraplégicos e tetraplégicos consigam andar. Disponível em: <<https://cuidarmais.wordpress.com/tag/paraplegia/>>. Acesso em 20 de agosto de 2017.

FARTAS, Felipe Costa. **Deep Learning para a Classificação de Sinais Cerebrais**. 2016.144 f. Dissertação (Mestrado em Engenharia da Computação) – Universidade de Pernambuco, Recife, 2016.

Ferreira, Erick da Penha; Alves, Nicholas de Lacerda. **Braço articulado com controle proporcional de movimento comandado via bluetooth por um aplicativo desenvolvido para plataformas android**. 41 f. São José dos Campos – SP.

FRANÇA, Rafael Ferreira. Indutor de Ondas Cerebrais por Batimento Binaural. 2008. 61 f. Dissertação (Engenharia da Computação) – Universidade Positivo/NCET, Curitiba, 2008.

GOMES, Marleide da Mota. Bases fisiológicas do eletroencefalograma. **Revista Brasileira de Neurologia**, Rio de Janeiro, v. 51, n. 1, jan - mar 2015.

HEMIPLEGIA: causas, sintomas e tratamento. Disponível em: <<http://fisioterapiamanual.com.br/blog/artigos/hemiplegia-causas-sintomas-tratamento/>>. Acesso em 20 de agosto de 2017.

HOW to hack toy EEGs. Disponível em: <<http://www.frontiernerds.com/brain-hack>>. Acesso em 10 de setembro de 2017.

INMOOV. Disponível em: <<http://inmoov.blogspot.com.br/>>. Acesso em 22 de outubro de 2017.

INTELIGÊNCIA Artificial, Educação de Crianças e o Cérebro Humano. Disponível em: <<http://www.intelliwise.com/reports/p4port.htm>>. Acesso em 27 de agosto de 2017.

INTRODUÇÃO as Redes Neurais Artificiais. Disponível em: <[http://www.gsigma.ufsc.br/~popov/aulas/rna/uteis/RNA\\_material\\_apoio.pdf](http://www.gsigma.ufsc.br/~popov/aulas/rna/uteis/RNA_material_apoio.pdf)>. Acesso em 04 de março de 2018.

LELIS, Atahualpa Bastos; Filho, Jarbas Cabral. **Utilização de ondas cerebrais para controle de componentes eletrônicos**. 31 f.

Levine, David M. Stephan, David F. Szabat, Kathryn A. Estatística – teoria e aplicação. Rio de Janeiro: 2016. 7° ed.

McVracken, Thomaz. Novo Atlas do corpo humano. São Paulo: Visor, 2001. 1v.

MG996R High Torque – Metal Gear Dual Ball Bearing Servo. Disponível em: <[http://www.electronicoscaldas.com/datasheet/MG996R\\_Tower-Pro.pdf](http://www.electronicoscaldas.com/datasheet/MG996R_Tower-Pro.pdf)>. Acesso em 17 de março de 2018.

Wecker, Jonas Edison. Músculos do antebraço. Disponível em: <<https://www.auladeanatomia.com/novosite/sistemas/sistema-muscular/musculos-do-membro-superior/musculos-do-antebraço/>>. Acesso em 07 de setembro de 2017.

MÚSCULOS do braço, antebraço e mão. Disponível em: <<http://www.anatomiadocorpo.com/sistema-muscular/musculos-do-braco-antebraço-e-mão/>>. Acesso em 07 de setembro de 2017.

MÚSCULOS do membro superior. Disponível em: <<http://anatomiaonline.com/musculos-do-membro-superior/>>. Acesso em 07 de setembro de 2017.

O que é Hemiplegia, quais são as causas, os sintomas e o tratamento. Disponível em: <<https://consultaremedios.com.br/crsaude/o-que-e-hemiplegia-quais-sao-as-causas-os-sintomas-e-o-tratamento/problemas-de-saude/sua-saude>>. Acesso em 20 de agosto de 2017.

PALKE, Amy. **Brainathlon: Enhancing Brainwave Control Through Brain – Controlled Game Play**. 2004. 49 f. Dissertação (Mestrado em Ciências da Computação Interdisciplinar) – Mills Coolege, 2004.

PARALISIA cerebral - Tecnologia e Acessibilidade. Disponível em: <<https://intervox.nce.ufrj.br/microfenix/adap.htm>>. Acesso em 20 de agosto de 2017.

PARALISIA cerebral e suas formas de apresentação clínica. Disponível em: <<https://interfisio.com.br/paralisia-cerebral-e-suas-formas-de-apresentacao-clinica-revisao-teorica/>>. Acesso em 20 de agosto de 2017.

PARAPLEGIA: o que é? Quais os tipos e as causas? Como é o tratamento? Disponível em: <<http://www.abc.med.br/p/348059/paraplegia+o+que+e+quais+os+tipos+e+as+causas+como+e+o+tratamento.htm>>. Acesso em 20 de agosto de 2017.

PENA, Alberto. Paraguaio cria braço biomecânico de baixo custo. Revista Exame, 2016. Disponível em: <<https://exame.abril.com.br/ciencia/paraguaio-cria-braco-biomecânico-de-baixo-custo/>>. Acesso em 20 de agosto de 2017.

PESQUISA pioneira cria software para pessoas com paralisia cerebral. Disponível em: <<http://www.fcee.sc.gov.br/index.php/sala-de-imprensa/noticias/8443-pesquisa-pioneira-no-estado-cria-software-para-comunicacao-de-pessoas-com-paralisia-cerebral>>. Acesso em 20 de agosto de 2017.

PRÓTESE. Disponível em: <<http://www.conforpes.com.br/proteses/>>. Acesso em 10 de setembro de 2017.

PRÓTESES biônicas. Disponível em: < <https://medium.com/tend%C3%AAncias-digitais/pr%C3%B3teses-bi%C3%B4nicas-a909e8da8834>>. Acesso em 20 de agosto de 2017.

QUAL é Monoplegia em Terminologia Médica? Disponível em: <<http://www.365saude.com.br/pt-conditions-treatments/pt-arthritis/1009030135.html>> Acesso em 20 de agosto de 2017.

REDE Neural. Disponível em: <<http://www.cerebromente.org.br/n05/tecnologia/plasticidade2.html>>. Acesso em 29 de outubro de 2017.

REDES neurais artificiais na classificação de frutos: cenário bidimensional. Disponível em: <<http://www.scielo.br/pdf/cagro/v27n2/a15v27n2.pdf>>. Acesso em 30 de junho de 2018.

SAIBA mais sobre a Síndrome do Membro Fantasma. Disponível em: <<http://www.leandroteles.com.br/blog/2013/03/17/saiba-mais-sobre-a-sindrome-do-membro-fantasma/>>. Acesso em 20 de agosto de 2017.

Silva, Ivan Nunes. Spatti, Danilo Hernane. Flauzino, Rogério Andrade. Redes neurais artificiais para engenharia e ciências aplicadas. São Paulo: 2016. 2º ed.

SÍNDROME do membro fantasma. Disponível em: <[www.leandroteles.com.br](http://www.leandroteles.com.br)>. Acesso em 20 de agosto de 2017.

TETRAPLEGIA e Paraplegia – causas e complicações em pessoas tetraplégicas e paraplégicas. Disponível em: < <https://www.copacabanarunners.net/tetraplegia.html>>. Acesso em 10 de setembro de 2017.

## ANEXO A – CÓDIGO DO ARDUINO

```
#include <Brain.h>

Brain brain (Serial);

void setup() {
  Serial.begin(9600); //Velocidade de transição de dados
}

void loop(){
  //Sensores Flex dos dedos
  int sensor_pol = analogRead(0);
  int sensor_ind = analogRead(1);
  int sensor_max = analogRead(2);
  int sensor_med = analogRead(3);
  int sensor_min = analogRead(4);

  if (brain.update()){

    Serial.print(brain.readCSV()); //Sinal do cérebro
    Serial.print(",");
    Serial.print(sensor_pol);
    Serial.print(",");
    Serial.print(sensor_ind);
    Serial.print(",");
    Serial.print(sensor_max);
    Serial.print(",");
    Serial.print(sensor_med);
    Serial.print(",");
    Serial.println(sensor_min);
  }
}
```

## ANEXO B – DATASHEET SENSOR FLEX



### FLEX SENSOR FS

#### Features

- Angle Displacement Measurement
- Bends and Flexes physically with motion device
- Possible Uses
  - Robotics
  - Gaming (Virtual Motion)
  - Medical Devices
  - Computer Peripherals
  - Musical Instruments
  - Physical Therapy
- Simple Construction
- Low Profile

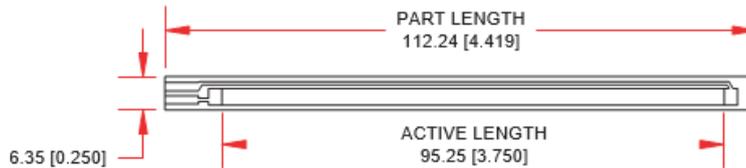
#### Mechanical Specifications

- Life Cycle: >1 million
- Height: 0.43mm (0.017")
- Temperature Range: -35°C to +80°C

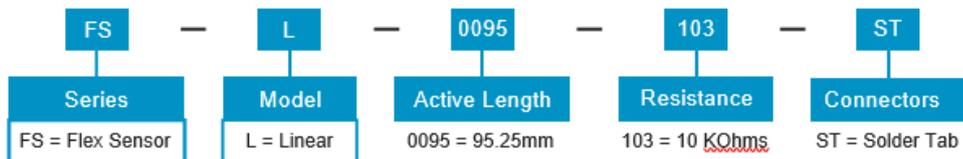
#### Electrical Specifications

- Flat Resistance: 10K Ohms
- Resistance Tolerance: ±30%
- Bend Resistance Range: 60K to 110K Ohms
- Power Rating: 0.50 Watts continuous. 1 Watt Peak

#### Dimensional Diagram - Stock Flex Sensor



#### How to Order - Stock Flex Sensor

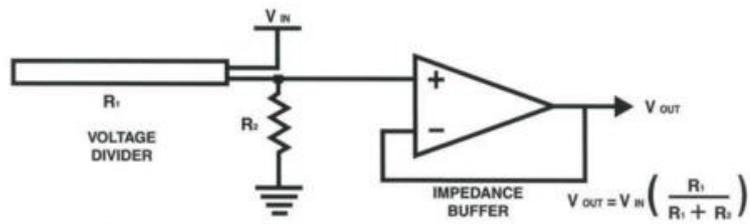


#### How It Works



Schematics

**BASIC FLEX SENSOR CIRCUIT:**

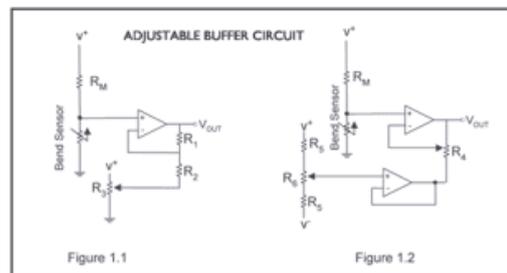


Following are notes from the ITP Flex Sensor Workshop

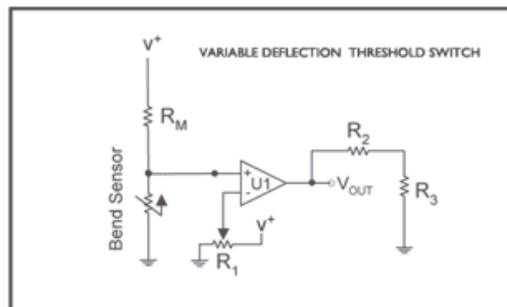
"The impedance buffer in the [Basic Flex Sensor Circuit] (above) is a single sided operational amplifier, used with these sensors because the low bias current of the op amp reduces error due to source impedance of the flex sensor as voltage divider. Suggested op amps are the LM358 or LM324."

"You can also test your flex sensor using the simplest circuit and skip the op amp."

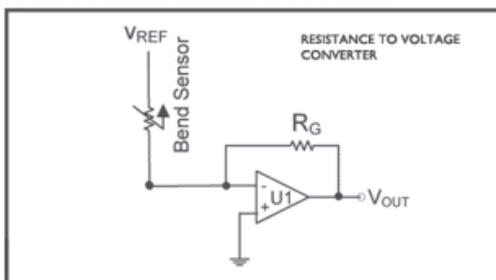
"Adjustable Buffer - a potentiometer can be added to the circuit to adjust the sensitivity range."



"Variable Deflection Threshold Switch - an op amp is used and outputs either high or low depending on the voltage of the inverting input. In this way you can use the flex sensor as a switch without going through a microcontroller."



"Resistance to Voltage Converter - use the sensor as the input of a resistance to voltage converter using a dual sided supply op-amp. A negative reference voltage will give a positive output. Should be used in situations when you want output at a low degree of bending."



## ANEXO C – CÓDIGO DE COLETA NO RASPBERRY

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <wiringPi.h>
#include <wiringSerial.h>
#include <unistd.h>
//Variáveis globais
int qualidade, atencao, meditacao, delta, teta, alfa_b, alfa_a,
beta_b, beta_a, gama_b, gama_a, polegar, indicador, max, medio, min;
void arquivo(char dados[128], char *msgNome){
char *s1, *s2, *s3, *s4, *s5, *s7, *s6, *s8, *s9, *s10, *s11, *s12,
*s13, *s14, *s15, *s16;
FILE *ax;
arqFinal = fopen(msgNome,"a"); //Cria um arquivo como os dados finais
s1=strtok(dados,","); //Separa os dados pelo caractere ','
s2=strtok(NULL,",");
s3= strtok(NULL,",");
s4=strtok(NULL,",");
s5=strtok(NULL,",");
s6=strtok(NULL,",");
s7=strtok(NULL,",");
s8=strtok(NULL,",");
s9=strtok(NULL,",");
s10=strtok(NULL,",");
s11=strtok(NULL,",");
s12=strtok(NULL,",");
s13=strtok(NULL,",");
s14=strtok(NULL,",");
s15=strtok(NULL,",");
s16=strtok(NULL,",");
qualidade=atoi(s1); //Transforma as dados de string para inteiro
atencao=atoi(s2);
meditacao=atoi(s3);
delta=atoi(s4);
teta=atoi(s5);
alfa_b=atoi(s6);

```

```

alfa_a=atoi(s7);
beta_b=atoi(s8);
beta_a=atoi(s9);
gama_b=atoi(s10);
gama_a=atoi(s11);
polegar=atoi(s12);
indicador=atoi(s13);
max=atoi(s14);
medio=atoi(s15);
min=atoi(s16);

fprintf(arqFinal, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d\n"
, qualidade, atencao, meditacao, delta, teta, alfa_b, alfa_a, beta_b, beta_a,
gama_b, gama_a, polegar, indicador, max, medio, min); //Escreve os valores
no arquivo

fclose(arqFinal);
}

int main(int arg, char **argv){ //O nome do arquivo é gerado ao chamar
o programa

    char *msgNome;
    int fd, d=0, msg, count, i, n, a=0;
    msgNome = argv[1];
    unsigned int nextTime;
    char message[128], c[2];
    size_t len;
    int countSerial;
    if ((fd = serialOpen ("/dev/ttyUSB0", 9600)) < 0){
        fprintf (stderr, "Unable to open serial device: %s\n", strerror
(errno));
        return 1;
    }
    if (wiringPiSetup () == -1){
        fprintf (stdout, "Unable to start wiringPi: %s\n", strerror
(errno));
        return 1;
    }
    nextTime = millis () + 300 ;
    for (count = 0; count <10; count++){
        for(n=0;n<128;n++){
            message[n]=' ';
        }
    }

```

```

message[127]='\0';
i=0;
countSerial = serialDataAvail(fd);
while (countSerial){
    delay(3);
    msg=serialGetchar(fd);
    if ((msg>47)&&(msg<58)&&(msg != 44)){
        sprintf(c,sizeof(c),"%c",msg);
        c[1]='\0';
        message[i]=c[0];
        i++;
    }
    else if (msg==44){
        message[i]=',';
        i++;
    }
    else{
        printf("%s\n",message);
        msg=serialGetchar(fd);
        arquivo(message,msgNome);
        fflush(stdout);
        delay(4);
        countSerial=serialDataAvail(fd);
        while (countSerial == 0){
            countSerial=serialDataAvail(fd);
        }
        for (n=0;n<128;n++)
            message[n]=' ';
        message[127]='\0';
        i=0;
    }
}
}
}

```

## ANEXO D – CÓDIGO DA RNA

```
#Bibliotecas
import os
import numpy as np
from sklearn.neural_network import MLPClassifier
import csv

#Abrir o arquivo de entrada
def ler_pro(arq):
    arq_aberto = open(arq, 'r')
    return csv.reader(arq_aberto, delimiter = ';')
arq = 'dados_entrada.csv'

#Abrir o arquivo de saida
dado_saida = csv.reader(open('dados_saida.csv', 'r'))

#Lista dos dados de entrada e saida
dados = ler_pro(arq)
entrada = []
saida = []

#Separação os valores de entrada
for linha_entrada in dados:
    linha_entrada = [float(i) for i in linha_entrada]
    entrada.append(linha_entrada)

#Separação os valores de saida
for linha_saida in dado_saida:
    saida.append(linha_saida)

#Classificação com base nos dados colocados no conjunto de dados
clf = MLPClassifier (solver='lbfgs', alpha=1e-5, hidden_layer_sizes
= (28,3), random_state=1)
hidden_layer_sizes = (28,3) - representação dos neurônios da camada
oculta da rede junto com o número de elementos de saída

clf.fit (entrada, saida)
```

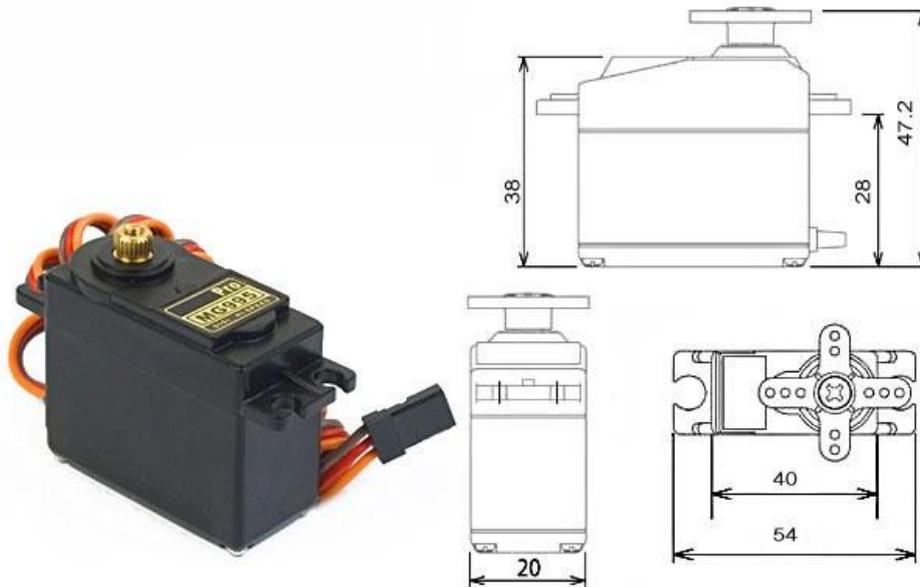
```
#'valor' é igual ao valor de saída da rede
valor=
(clf.predict([[0.6875,0.34,0.37,0.320864032,0.063046211,0.04573347,0
.030939715,0.111963316,0.125595107,0.118733781,0.145373806]]))

print(valor)
```

## ANEXO E – DATASHEET SERVO MOTOR MG995

### MG995 High Speed

#### Metal Gear Dual Ball Bearing Servo



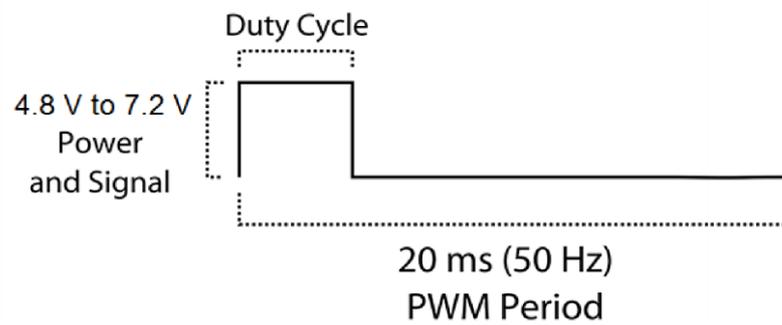
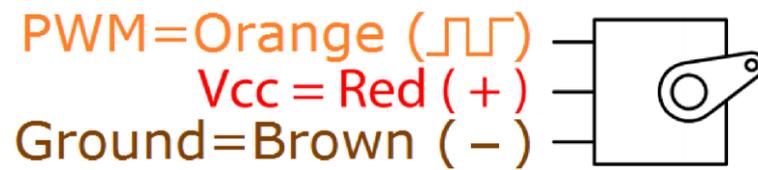
The unit comes complete with 30cm wire and 3 pin 'S' type female header connector that fits most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-speed standard servo can rotate approximately 120 degrees (60 in each direction). You can use any servo code, hardware or library to control these servos, so it's great for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG995 Metal Gear Servo also comes with a selection of arms and hardware to get you set up nice and fast!

#### Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 8.5 kgf·cm (4.8 V ), 10 kgf·cm (6 V)

- Operating speed: 0.2 s/60° (4.8 V), 0.16 s/60° (6 V)
- Operating voltage: 4.8 V a 7.2 V
- Dead band width: 5  $\mu$ s
- Stable and shock proof double ball bearing design
- Temperature range: 0 °C – 55 °C



## ANEXO F – CÓDIGO DE ATUAÇÃO DOS MOTORES

```

#include <stdio.h>
#include <unistd.h>
#include <wiringPi.h>
#define MAX_TIME 50
void servo_pwm1(int servo1, int servo2, int value){
    int n;
    for (n=0; n<MAX_TIME; n++){
        digitalWrite (servo1, 1); // On
        digitalWrite (servo2, 1);
        usleep(value);
        digitalWrite (servo1, 0); // Off
        digitalWrite (servo2, 0);
        usleep(20000-value);
    }
}

void servo_pwm2(int servo3, int servo4, int servo5, int value){
    int n;
    for (n=0; n<MAX_TIME; n++){
        digitalWrite (servo3, 1); // On
        digitalWrite (servo4, 1);
        digitalWrite (servo5, 1);
        usleep(value);
        digitalWrite (servo3, 0); // Off
        digitalWrite (servo4, 0);
        digitalWrite (servo5, 0);
        usleep(20000-value);
    }
}

int main (void){
    int n, i, teste, pwm_value;
    int servo1_pin=0; //Servo do dedo indicador
    int servo2_pin=1; //Servo do dedo médio
    int servo3_pin=2; // Servo do dedo anular
    int servo4_pin=3; // Servo do dedo mínimo
    int servo5_pin=4; // Servo do dedo polegar

```

```
if (wiringPiSetup () == -1)
    return 1;
pinMode (servo1_pin, OUTPUT); //BCM_GPIO0 pin 11
pinMode (servo2_pin, OUTPUT); //BCM_GPIO1 pin 12
pinMode (servo3_pin, OUTPUT); //BCM_GPIO2 pin 13
pinMode (servo4_pin, OUTPUT); //BCM_GPIO3 pin 15
pinMode (servo5_pin, OUTPUT); //BCM_GPIO4 pin 16
pwm_value=1500;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=500;
servo_pwm1(servo1_pin, servo2_pin, pwm_value);
pwm_value=1300;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=700;
servo_pwm1(servo1_pin, servo2_pin, pwm_value);
pwm_value=1100;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=900;
servo_pwm1(servo1_pin, servo2_pin, pwm_value);
pwm_value=900;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=1100;
servo_pwm1(servo1_pin, servo2_pin, pwm_value);
pwm_value=700;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=1300;
servo_pwm1(servo1_pin, servo2_pin, pwm_value);
pwm_value=500;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=1500;
servo_pwm1(teste, servo2_pin, pwm_value);
pwm_value=700;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=1300;
servo_pwm1(teste, servo2_pin, pwm_value);
pwm_value=900;
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);
pwm_value=1100;
```

```
servo_pwm1(teste, servo2_pin, pwm_value);  
pwm_value=1100;  
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);  
pwm_value=900;  
servo_pwm1(teste, servo2_pin, pwm_value);  
pwm_value=1300;  
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);  
pwm_value=700;  
servo_pwm1(teste, servo2_pin, pwm_value);  
pwm_value=1500;  
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);  
pwm_value=500;  
servo_pwm1(teste, servo2_pin, pwm_value);  
pwm_value=1700;  
servo_pwm2(servo3_pin, servo4_pin, servo5_pin, pwm_value);  
return 0;
```