



Universidade de Brasília – UnB  
Faculdade UnB Gama – FGA  
Engenharia Eletrônica

# **Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina**

**Autores:** Heleno da Silva Moraes, Oziel da Silva Santos  
**Orientador:** Dr, Marcus Vinícius Chaffim Costa

Brasília, DF  
2017





Heleno da Silva Moraes, Oziel da Silva Santos

# **Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina**

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Dr, Marcus Vinícius Chaffim Costa

Coorientador: PhD, Daniel Mauricio Muñoz Arboleda

Brasília, DF

2017

---

Heleno da Silva Morais, Oziel da Silva Santos

Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina/ Heleno da Silva Morais, Oziel da Silva Santos. – Brasília, DF, 2017-

48 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr, Marcus Vinícius Chaffim Costa

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB  
Faculdade UnB Gama – FGA , 2017.

1. Sistemas em Chip. 2. LDA. I. Dr, Marcus Vinícius Chaffim Costa. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina

CDU 02:141:005.6

---

# Errata



Heleno da Silva Moraes, Oziel da Silva Santos

# **Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina**

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Trabalho aprovado. Brasília, DF, 11 de Dezembro de 2017 – Data da aprovação do trabalho:

---

**Dr, Marcus Vinícius Chaffim Costa**  
Orientador

---

**Dr, Cristiano Jacques Miosso  
Rodrigues Mendes**  
Convidado 1

---

**Dr, Renato Coral Sampaio**  
Convidado 2

Brasília, DF  
2017





**Heleno da Silva Moraes**

*Ao meu pai, Joaquim Albino de Moraes †*

Oziel da Silva Santos

*Aos meus pais, José Antônio e Gezonita Maurício.*

# Agradecimentos

**Heleno da Silva Moraes**

Agradeço primeiramente a Deus, por há 6 anos atrás ter escutado o pedido de um jovem sonhador, e que ao longo de toda minha vida esteve ao meu lado, mesmo quando não mereci.

Agradeço à toda minha família, minha mãe e meus irmãos que sempre me motivaram para conquistar meus sonhos. À todos os meus amigos, em especial Ebenezer Andrade e Oziel da Silva, que estiveram presentes nos bons e maus momentos desde o início dessa jornada e a todos que contribuíram de alguma forma durante estes 6 anos para a realização deste sonho pessoal, mas que faz parte do sonho de muitos envolvidos. À minha namorada, que desde o primeiro dia tornou desse meu sonho, nosso sonho, e que me apoiou em todos os momentos.

Aos meus orientadores pela paciência e dedicação por sempre estarem a disposição para a realização deste trabalho.

E por fim, mas não menos importante um agradecimento em especial para minha madrinha e sogra, Maria, que esteve presente nos momentos mais difíceis financeiramente ou emocionalmente. Sem ela não poderia estar aqui.

## Oziel da Silva Santos

Aqui tem vários nomes de quem apoiou e de uma forma direta ou indireta colaborou para esse trabalho, ainda que não houvesse espaço, teria que citar: O nome de Deus que me concedeu a oportunidade de cursar Engenharia Eletrônica nesta universidade. Agradecer aos meus pais pelo apoio e amor sem fim. Aos meus orientadores, professor e professor pelo grande interesse em ajudar e ensinar. A minha namorada Brunna Siqueira, que sempre esteve ao meu lado. Aos meus amigos: Heleno da Silva, Ebenezer Andrade, Pedro Ivo, Ithallo Junior, Lucas Raposo, Gabriel Henrique, Gabriela Volpato, pelas suas ideias, conselhos e ajudas. Aos autores que são base teórica para este trabalho, em especial Fabien Lotte. E à esta Universidade.

# Resumo

As aplicações com as *Brain Computer Interfaces* (BCI) apresentam um desenvolvimento crescente graças ao aumento do interesses de pesquisadores sobre o tema. Um dos principais passos para desenvolvimento de uma BCI é a classificação dos sinais cerebrais, que posteriormente são convertidos em comandos de controle para um dispositivo. Alguns classificadores apresentam características de linearidade, não exigindo um alto esforço computacional para sua execução, o que possibilita a sua implementação em um sistema embarcado. Sendo assim, este trabalho apresenta um estudo comparativo entre as implementações do algoritmo de treinamento do classificador *Linear Discriminant Analysis* (LDA), em hardware e em software, utilizando o SoC da família *Zynq* embarcado no kit de desenvolvimento *Zybo-board*. **Palavras-chaves:** BCI; LDA; FPGA; SoC; Sistemas Embarcados.



# Abstract

*The Brain Computer Interface (BCI) applications present a growing development due to increase researchers interesting about the subject. One of the main step in BCI development is the brain signals classification, wich are then translate into controls commands for any device. Some classifiers have characteristics of linearity, not requaring a high computational performance for their execution, wich allow its implementation in embedded systems. Thus, this work present a comparative study between implementation of the Linear Discriminant Analysis (LDA) classifier training algorithm, in harware and software, using the SoC of the Zynq family embedded in the development kit Zybo-board.*

**Key-words:** BCI; LDA; FPGA; SoC; Embedded Systems.





# Lista de ilustrações

Figura 1 – Três principais áreas do cérebro. Fonte: (ALVAREZ; LEMOS, 2006) . . .	29
Figura 2 – Estrutura de um neurônio. Fonte: (SIULY, 2012) . . . . .	30
Figura 3 – Sistema Internacional de Posicionamento 10-20. Fonte: (CAMPISI; ROCCA; SCARANO, 2012). . . . .	32
Figura 4 – Exemplos de diferentes tipos de EEG. Fonte: (CAMPISI; ROCCA; SCARANO, 2012) . . . . .	33
Figura 5 – Diferença entre os sistemas ECoG e EEG. Fonte: (WOLPAW, 2012). .	34
Figura 6 – Fluxograma de processos de uma BCI, onde pode ser aplicado em mecanismos motorizados, reabilitação muscular, alarmes, entre outros (Adaptado de (WOLPAW, 2012)). . . . .	34
Figura 7 – Placa de Desenvolvimento Zybo-Board. Fonte: (DIGILENT, 2017) . . .	39
Figura 8 – Fluxograma de implementação de hardware em um SoC. Adaptado de (CROCKETT et al., 2014) . . . . .	42



# Lista de tabelas

Tabela 1 – Número de tarefas rotuladas e não rotuladas por sujeito (BLANKERTZ et al., 2006). . . . .	36
Tabela 2 – Acurácia de classificação do algoritmo LDA utilizando do algoritmo CSP para maximização de variância de classes. (LOTTE; GUAN, 2011)	40
Tabela 3 – Cronograma de atividades já desenvolvidas para este presente trabalho	44
Tabela 4 – Cronograma de atividades a serem desenvolvidas para este presente trabalho. . . . .	45



# Lista de abreviaturas e siglas

BCI	<i>Brain Computer Interface</i>
EEG	Eletroencefalograma
SVM	<i>Support Vector Machine</i>
LDA	<i>Linear Discriminant Analysis</i>
SoC	<i>System on Chip</i>
FPGA	<i>Field Programmable Array</i>
SNC	Sistema Nervoso Central
VHDL	<i>VHISC Hardware Description Language</i>
VHISC	<i>Very High Speed Integrated Circuit</i>
LUT	<i>Look Up Table</i>
FF	Flip-Flop
DSP	<i>Digital Signal Processor</i>
RAM	<i>Random Access Memory</i>
MUX	Multiplexador
CSP	<i>Common Spatial Pattern</i>
ECoG	Eletrocorticografia



# Lista de símbolos

$\mu$	Micro
$\theta$	<i>Theta</i>
$\alpha$	<i>Alpha</i>
$\beta$	<i>Beta</i>
$\gamma$	<i>Gamma</i>





# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>25</b>
<b>1.1</b>	<b>Contextualização</b>	<b>25</b>
<b>1.2</b>	<b>Justificativa</b>	<b>27</b>
<b>1.3</b>	<b>Objetivos</b>	<b>27</b>
1.3.1	Objetivos Gerais	27
1.3.2	Objetivos Específicos	27
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>29</b>
<b>2.1</b>	<b>O Cérebro</b>	<b>29</b>
<b>2.2</b>	<b>Eletroencefalografia</b>	<b>31</b>
<b>2.3</b>	<b><i>Brain Computer Interface</i></b>	<b>33</b>
<b>2.4</b>	<b><i>BCI Competition</i></b>	<b>35</b>
2.4.1	<i>BCI Competition III</i>	36
2.4.2	<i>BCI Competition III - Dataset IVa</i>	36
<b>2.5</b>	<b><i>Linear Discriminant Analysis</i></b>	<b>37</b>
<b>2.6</b>	<b><i>System-on-Chip</i></b>	<b>38</b>
2.6.1	Aquitetura Simplificada de um SoC	38
2.6.2	<i>Zybo-Board</i>	39
<b>2.7</b>	<b>Estado da Arte</b>	<b>39</b>
<b>3</b>	<b>PROPOSTA METODOLÓGICA</b>	<b>41</b>
<b>3.1</b>	<b>Implementação em hardware</b>	<b>41</b>
3.1.1	Dispositivos e ferramentas	41
3.1.2	Metodologias de desenvolvimento	41
<b>3.2</b>	<b>Implementação em software</b>	<b>42</b>
3.2.1	Materiais	43
3.2.2	Métodos e Técnicas	43
<b>3.3</b>	<b><i>Data Set IVa</i></b>	<b>43</b>
<b>3.4</b>	<b>Cronograma de Atividades</b>	<b>43</b>
	<b>REFERÊNCIAS</b>	<b>47</b>



# 1 Introdução

Este capítulo apresenta uma visão geral do projeto, apresentando sua contextualização, assim como a proposta do presente trabalho e seus objetivos a serem alcançados.

## 1.1 Contextualização

Através de uma rede de mais de 100 bilhões de células nervosas interconectadas, o cérebro realiza o controle de nossas ações, percepções, emoções e etc (KANDEL, 2013). Estas células são chamadas de *neurônios*, e neles são armazenados sinais elétricos, que representam todas as informações de controle (SIULY; LI; ZHANG, 2017). Estes sinais podem ser medidos pela eletroencefalografia (EEG), que é um sistema de medição de sinais elétricos produzidos pelo cérebro durante atividades cerebrais (LOTTE; GUAN, 2011). Segundo (SIULY, 2012), a EEG é uma das mais importantes ferramentas para diagnosticar doenças cerebrais.

Além do diagnóstico de doenças cerebrais uma outra aplicação para os sinais adquiridos pela EEG são as *Brain Computer Interfaces* (BCIs) (LOTTE; GUAN, 2011). Uma BCI é um sistema que realiza a comunicação entre o cérebro e um computador (SIULY; LI; ZHANG, 2017), onde sua principal função é a tradução dos sinais elétricos cerebrais em comandos de controle para qualquer dispositivo eletrônico (SIULY; LI; ZHANG, 2017).

A BCI realiza a tradução destes sinais através de seis passos: 1) medição dos sinais provenientes de atividades cerebrais, normalmente através da EEG, 2) pré-processamento destes sinais, 3) extração de características, 4) classificação, 5) tradução dos sinais em comandos e 6) realimentação (MASON; BIRCH, 2003). Um dos principais passos para a implementação de uma BCI é a **classificação**, pois é após este passo que é realizada a tradução dos sinais provenientes da EEG em comandos de controle (MASON; BIRCH, 2003).

A classificação de um sinal é caracterizada, em aprendizado de máquina e em reconhecimento de padrões, como um algoritmo que atribui parte de um dado sinal de entrada a um dado número de classes ou categorias (BRUNELLI, 2009). Um exemplo é a classificação de um e-mail como "spam" ou "não-spam". Os algoritmos que realizam a classificação dos sinais de entrada são chamados de **classificadores** (SIULY; LI; ZHANG, 2017). De acordo com (LOTTE, 2008, p. 41), "estes classificadores são capazes de aprender como identificar um vetor de características, graças aos processos de treinamentos".

O algoritmo que realiza a classificação é caracterizado por uma função matemática

que mapeia um sinal de entrada em sua respectiva classe (LOTTE, 2008). Os classificadores preferidos pelos pesquisadores são os **classificadores supervisionados**, pois estes tipos de classificadores necessitam de um conjunto de dados de treinamento para realizar o processo de treinamento do classificador. O conjunto de dados para realização do processo de treinamento são formados por vetores de características previamente atribuídos às suas respectivas classes (LOTTE, 2008). Portanto os classificadores supervisionados são implementados a partir de dois processos: **treinamento** e **testes** (SIULY; LI; ZHANG, 2017). As *Support Vector Machines* (SVM), os *Linear Discriminant Analysis* (LDA), os filtros Kalman e as árvores de decisões são alguns exemplos mais conhecidos de classificadores do tipo supervisionados (SIULY; LI; ZHANG, 2017).

O LDA como dito anteriormente é um dos classificadores supervisionados e tem como principais vantagens a simplicidade e atratividade computacional, por se tratar de um classificador linear (THEODORIDIS; KOUTROUMBAS et al., 1999). O objetivo do LDA é usar uma transformação linear para encontrar um conjunto otimizado de vetores discriminantes e remapear o conjunto de características original, em um outro conjunto de dimensão inferior (SHASHOA et al., 2016).

A simplicidade e robustez do LDA possibilita sua implementação em um sistema embarcado, o que viabiliza sua acessibilidade a qualquer usuário, um exemplo de sistema a se embarcar este tipo de algoritmo são os *Systems on Chip* (SoC).

Um SoC é caracterizado pela implementação de todo um sistema computacional, tais como memórias, processadores, entradas e saídas, lógicas digitais, entre outros, em um único chip de silício (CROCKETT et al., 2014). Diferente dos computadores tradicionais, que possuem seu sistema implementado a partir de módulos isolados e combinados em uma placa de circuito impresso, ou placa-mãe, os SoCs possuem como principais características um baixo custo de implementação, além de baixo consumo de potência, menor tamanho físico, maior confiabilidade e maior velocidade do sistema geral, quando comparado com um computador tradicional (CROCKETT et al., 2014). Um exemplo de um SoC é a plataforma *Zynq* que combina em um único chip processadores *Advanced Risc Machine* (ARM) e *Field Programmable Gate Array* (FPGA), esse último utilizado para configurar todos os módulos de um computador tradicional (CROCKETT et al., 2014).

Tendo em vista a grande vantagem dos SoCs sobre os computadores tradicionais, onde são implementados e executados os algoritmos de classificação, este trabalho apresenta um estudo da viabilidade da implementação em hardware e em software embarcado, do algoritmo de treinamento do classificador LDA desenvolvido previamente por (LOTTE; GUAN, 2011) na plataforma *Matlab*, realizando a comparação de consumo computacional, processamento computacional (tempo de execução), entre as implementações em hardware e software, onde a implementação em hardware consiste no mapeamento do algoritmo na FPGA da plataforma *Zynq*, afim de paralelizar seus processos, já a implementação em

software consiste em executar o algoritmo em um sistema embarcado utilizando os cores ARM, também da plataforma *Zynq*, além da comparação com sua implementação inicial em *Matlab*.

## 1.2 Justificativa

As aplicações das BCIs apresentam um crescente desenvolvimento, graças ao aumento do interesse em pesquisas voltadas para o tema (BLANKERTZ et al., 2006). Por ser considerado o principal processo das BCIs, a classificação requer um cuidado especial (MASON; BIRCH, 2003). Como o LDA é um classificador supervisionado, a acurácia da classificação depende inteiramente de um bom treinamento (LOTTE; GUAN, 2011). Por se tratar de um algoritmo linear e consequentemente não exigir um grande esforço computacional, torna-se viável sua implementação em um sistema embarcado. Como os SoCs apresentam características de baixo consumo de potência, tamanho físico pequeno e possuir todos módulos de um sistema computacional em um único chip, a implementação de algoritmos de classificação em sistema deste porte podem tornar as BCIs mais acessíveis, tendo em vista que um algoritmo de treinamento embarcado em um SoC reduzirá a necessidade de um sistema computacional tradicional, além de um melhor processamento computacional, pois sua implementação em plataformas como FPGAs torna-se possível paralelizar seus processos de execução (CROCKETT et al., 2014).

## 1.3 Objetivos

Esta seção apresenta os objetivos gerais e específicos propostos a serem desenvolvidos neste presente trabalho.

### 1.3.1 Objetivos Gerais

- Implementar parte do algoritmo de treinamento de um classificador LDA em um sistema embarcado, utilizando um SoC na plataforma *Zynq* afim de otimizar o algoritmo, tanto em relação a tempo de execução, quanto o seu consumo de recursos.

### 1.3.2 Objetivos Específicos

- Explorar o algoritmo de treinamento do classificador LDA desenvolvido por (LOTTE; GUAN, 2011);
- Mapear parte deste algoritmo em arquiteturas paralelas (arquiteturas de hardware) utilizando a linguagem VHDL;

- Implementar em sistema embarcado o algoritmo de treinamento utilizando os cores ARM da *Zynq*;
- Realizar teste e validação das implementações utilizando as bases de dados do BCI Competition III, em específico o conjunto de dados BCI III *dataset IVa*.
- Realizar uma análise estatística do erro associado a ambas implementações, comparadas com a implementação na plataforma *Matlab*.

## 2 Referencial Teórico

Este capítulo apresenta os conceitos teóricos que abordam este trabalho, detalhando a estrutura cerebral responsável pela geração dos sinais de interesse na subseção 2.1, o sistema de captura dos sinais (a EEG) na subseção 2.2, o sistema que se refere às BCIs que realizam a tradução dos sinais em comandos para dispositivos na subseção 2.3, os detalhes técnicos a respeito dos sinais oferecidos pela base de dados *BCI Competition* na subseção 2.4, a estrutura geral de um classificador LDA na subseção 2.5, os conceitos de um SoC na subseção 2.6 e por fim o estado da arte das implementações de algoritmos de classificação na subseção 2.7.

### 2.1 O Cérebro

O Sistema Nervoso Central (SNC) é o responsável direto pelo comando do nosso comportamento geral (CLARK NASHAAT BOUTROS, 2005). Ele pode ser dividido em três principais áreas: tronco encefálico ou medula espinhal, o cérebro e o cerebelo (Figura 1) (ALVAREZ; LEMOS, 2006).

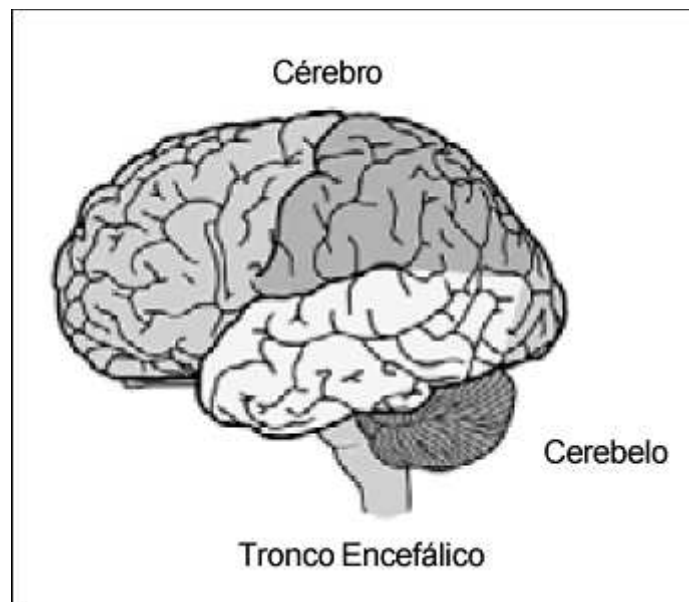


Figura 1 – Três principais áreas do cérebro. Fonte: (ALVAREZ; LEMOS, 2006)

O tronco encefálico, parte caudal do SNC recebe e processa todos os sinais dos sensores corporais, além de realizar o controle dos membros e do tronco humano (KANDEL, 2013).

O cérebro é o processador do SNC, nele são recebidos e processados os sinais da

medula espinhal, além de fornecer todos os sinais de controle para a própria medula, que por sua vez distribui os sinais para os membros e tronco (KANDEL, 2013).

O cerebelo é localizado logo atrás do tronco encefálico que se conectam através de fibras chamadas de pedúnculos (KANDEL, 2013), é a segunda maior estrutura do SNC contendo mais da metade dos neurônios cerebrais, (SIULY, 2012) é responsável pelo mapeamento em volta do indivíduo, ou seja, sua percepção (ALVAREZ; LEMOS, 2006), além do controle da região motora e a memória de movimentos (SIULY, 2012; ALVAREZ; LEMOS, 2006)

Os sinais de controle e de sensoriamento são sinais elétricos armazenados nos neurônios (KANDEL, 2013). Suas características eletroquímicas permitem que os neurônios armazenem e transmitam sinais elétricos para qualquer outra célula receptora mesmo que a longa distâncias (SIULY, 2012). Sua estrutura pode ser dividida em três principais partes: 1) Corpo da célula, 2) axônio e 3) dendrito, conforme apresentado na Figura(2).

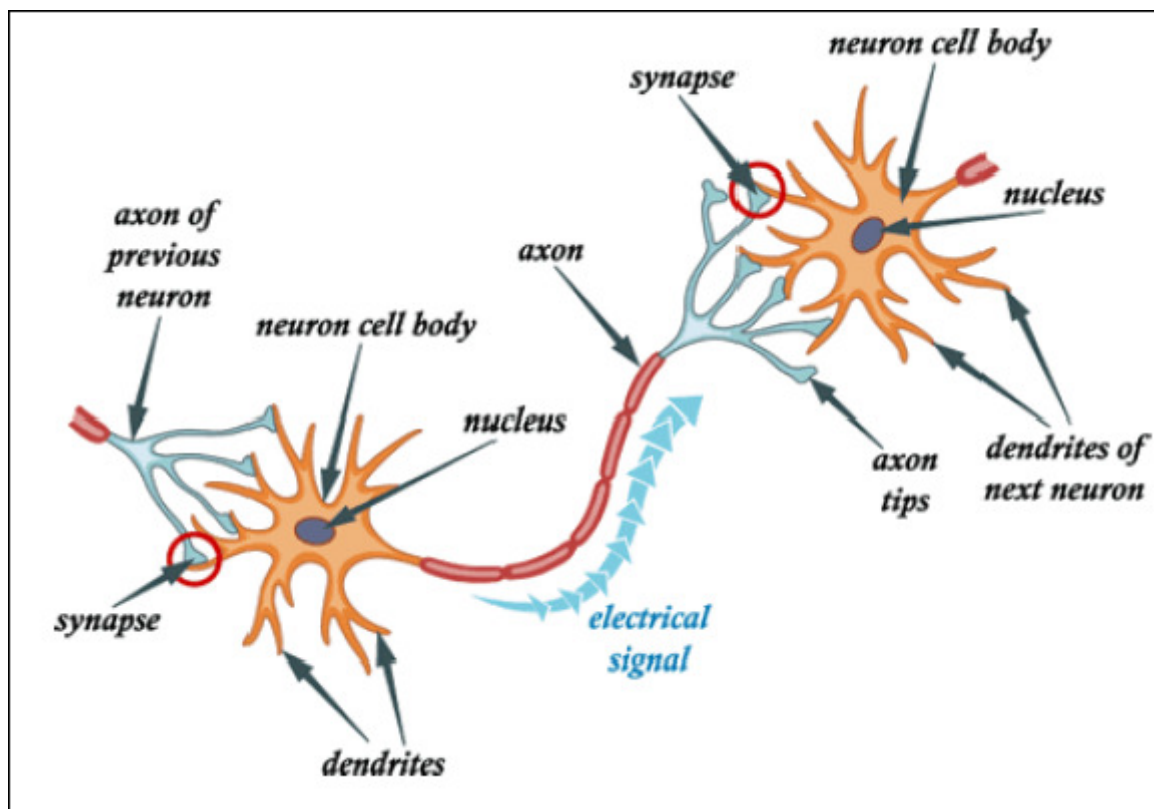


Figura 2 – Estrutura de um neurônio. Fonte: (SIULY, 2012)

A ativação dos neurônios é dada através de um gradiente de concentração eletroquímico, resultando na produção de uma corrente elétrica cerebral que flui através do axônio, o que torna possível a comunicação com outras células (SIULY, 2012). A corrente elétrica cerebral consiste comumente de íons de  $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{Ca}^{++}$  e  $\text{Cl}^-$  (TEPLAN et al., 2002).



As atividades elétricas cerebrais podem ser divididas em dois conjuntos: os potenciais de ação (AP) e os potenciais de sinapse (SP) (SIULY, 2012).

Sempre que um potencial de sinapse atinge o limite de condução, ou seja, se o potencial é carregado (estimulado por alguma atividade cerebral) até o ponto em que se é gerada uma corrente elétrica no axônio, um AP é iniciado (SIULY, 2012).

Como dito antes, o cerebelo é o responsável pelo controle da percepção e dos movimentos, além da memória de movimento, portanto a percepção, os movimentos e a imagética motora são os estímulos que iniciam uma AP na região do cerebelo (ALVAREZ; LEMOS, 2006).

Os potenciais elétricos gerados a partir dos potenciais de sinapse são armazenados na escalpe (ou couro cabeludo), onde durante uma atividade cerebral é criado um dipolo elétrico entre os dendritos e a soma (região ao redor do núcleo) (SIULY, 2012). Estes potenciais podem ser medidos pela eletroencefalografia, conteúdo abordado na seção 2.2 deste trabalho.

## 2.2 Eletroencefalografia

A Eletroencefalografia (EEG) é um sistema de aquisição de potenciais elétricos, que refletem atividades cerebrais humano (SIULY; LI; ZHANG, 2017). É muito usado por profissionais de saúde e cientistas para avaliar e estudar funções cerebrais e diagnosticar distúrbios neurológicos (SIULY; LI; ZHANG, 2017).

A técnica popular que registra sinais do cérebro usando eletrodos estrategicamente colocados sobre o couro cabeludo do paciente, portanto não invasiva, é muito importante nos diagnósticos de doenças neurológicas (RAO, 2013). Em um exame de EEG, dependendo da sua aplicação, podem ser usados de 1 a 256 eletrodos que iram captar os sinais de forma paralela (RAO, 2013). Cada par de eletrodos formam um canal, portanto com 256 eletrodos tem-se uma leitura de 128 canais, também conhecido como EEG multi canais (RAO, 2013). Cada canal recebe um amplificador de instrumentação e um equipamento de gravação do EEG (RAO, 2013).

Devido às diferentes camadas e tecidos interpostos entre a fonte do sinal (atividade neural no córtex) e os sensores colocados no couro cabeludo, tudo isso atuando como um filtro passa-baixa, faz com o que a resolução espacial do EEG seja ruim, em contra partida tem boa resolução temporal na faixa de milisegundos (RAO, 2013).

Em um adulto normal, o sinal típico de EEG tem sua amplitude variando de 1 a 100 microvolt, se for medido usando eletrodos de tipo agulha o seu módulo pode variar de 10 a 20 milivolt. Considera a não uniformidade do cérebro humano e a organização funcional do cortex, o sinal de EEG pode variar bastante de acordo com a disposição

dos eletrodos (SIULY, 2012).

Com a baixa amplitude desse tipo de sinal, ele pode sofrer facilmente atenuações, contaminações em seu espectro, como por exemplo interferência da rede elétrica de 60 Hz e seus harmônicos associados, e principalmente atividades musculares exercidas no ato da extração dos sinais. Portanto no momento do exame os pacientes são orientados a não realizar nenhum tipo de movimento (RAO, 2013).

Para especificar um padrão de alocação dos eletrodos na cabeça dos pacientes ou voluntários, é usado o sistema internacional 10-20, esse padrão determina as distancias entre eletrodos (SIULY; LI; ZHANG, 2017). São posicionados eletrodos primeiramente em pontos estratégicos, os pontos são nomeados por: *mastoids*, eletrodos de referência localizados atrás de cada orelha (A1 e A2), *Nasion*, referência no segmento do topo do nariz, porém nivelado com os olhos e por fim, o *onion*, referência situada na base do crânio no ponto médio da parte de trás da cabeça, as medidas são feitas a partir desses pontos, no sentido transversal e médio, com intervalos de 10 e 20 % (RAO, 2013), conforme Figura 3.

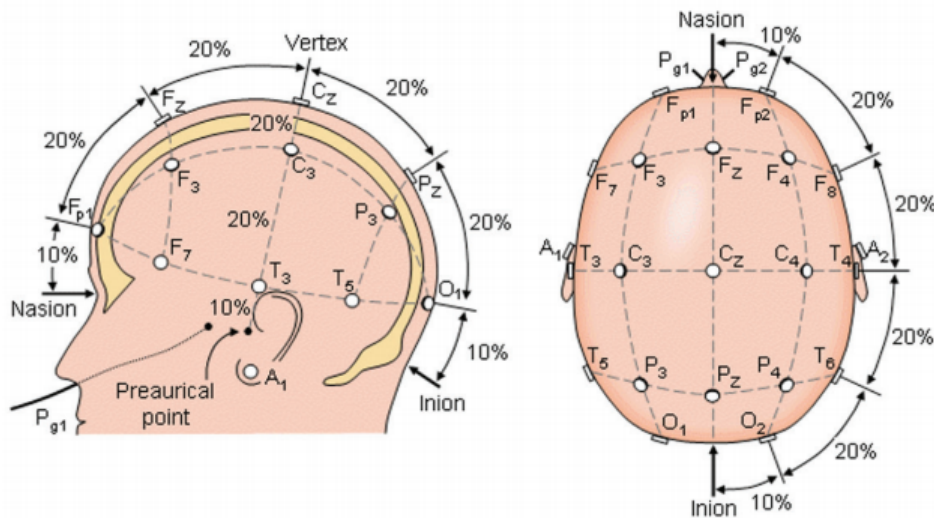


Figura 3 – Sistema Internacional de Posicionamento 10-20. Fonte: (CAMPISI; ROCCA; SCARANO, 2012).

A Frequência do sinal é um dos parâmetros mais importantes na avaliação clínica de anomalias em EEGs, e também para compreender o comportamento operacional na pesquisa cognitiva (SIULY, 2012). Com inúmeras oscilações não periódicas milhares de comunicação entre neurônios, comportamentos imprevisíveis o EEG humano é comumente ordenado em faixas de frequências específicas, as bandas são divididas em: delta (0.5-4 Hz), theta( $\theta$ )(4-8Hz), alpha( $\alpha$ )(8-13Hz), beta( $\beta$ )(13-30Hz) e acima de 30Hz gama( $\gamma$ ), conforme Figura 4.

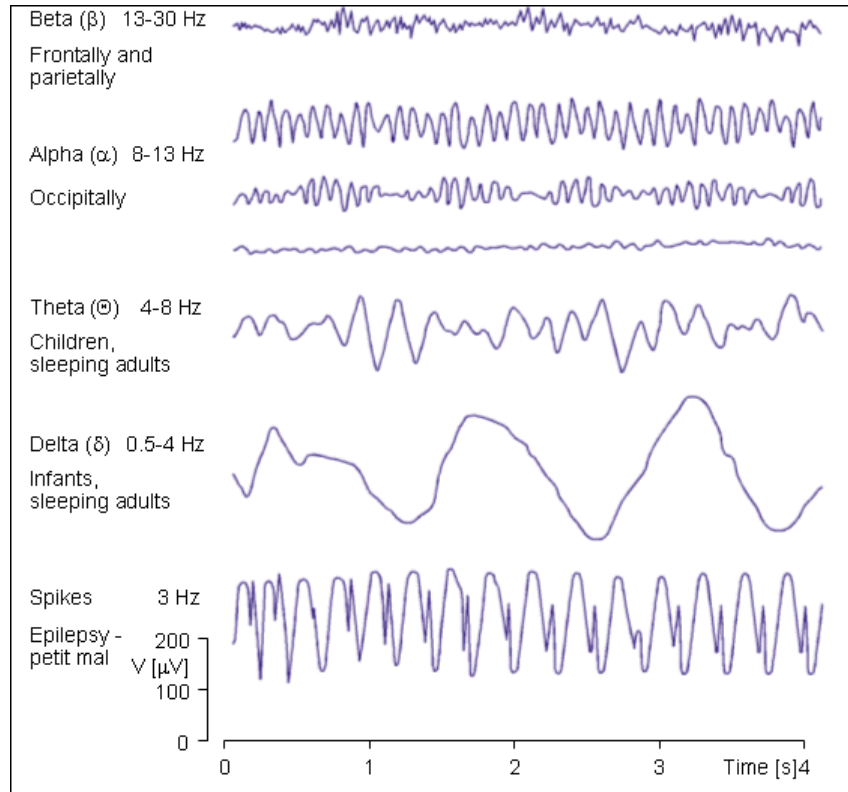


Figura 4 – Exemplos de diferentes tipos de EEG. Fonte: (CAMPISI; ROCCA; SCARANO, 2012)

Uma das aplicações dos sinais adquiridos pela EEG são as *Brain Computer Interface* (LOTTE; GUAN, 2011; SIULY, 2012), que é apresentado na seção 2.3.

## 2.3 Brain Computer Interface

Uma *Brain Computer Interface* (BCI) (em português, Interface Cérebro-Máquina) é um sistema que analisa sinais cerebrais e os convertem para um novo sinal de saída (WOLPAW, 2012), ou seja, uma BCI é um sistema que realiza a comunicação entre o cérebro e o mundo externo sem a interação neuromuscular (WOLPAW, 2012). O termo BCI foi utilizado pela primeira vez por Jacques Vidal em 1970, onde foram utilizadas técnicas invasivas para aquisição dos sinais cerebrais, sistema de eletrocorticografia (ECoG) (WOLPAW, 2012), onde os sensores de aquisição eram instalados na região do cortex. Em 1980 o mesmo Jacques Vidal publicou pela primeira vez a utilização de uma BCI não invasiva utilizando o sistema da EEG (GUGER et al., 2013), as duas técnicas são apresentadas na Figura 5.

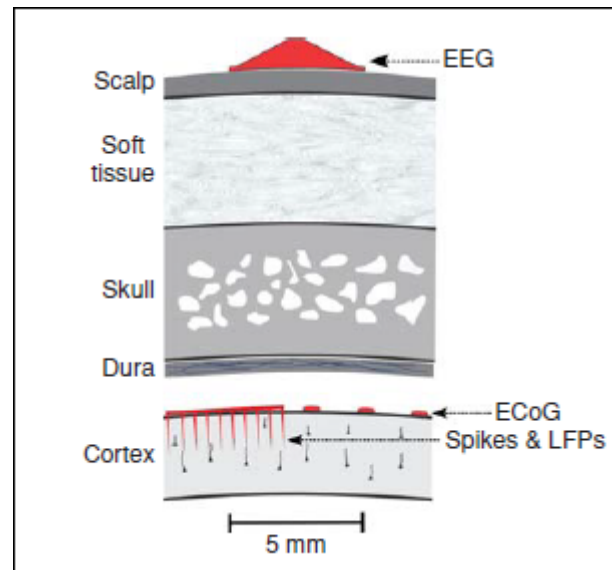


Figura 5 – Diferença entre os sistemas ECoG e EEG. Fonte: (WOLPAW, 2012).

Atualmente é mais comum a utilização do sistema da EEG para aquisição de sinais cerebrais por se tratar de um sistema não invasivo (GUGER et al., 2013). A BCI realiza a tradução dos sinais cerebrais através de seis passos: 1) medição dos sinais cerebrais, 2) pré-processamento destes sinais, 3) extração de características, 4) classificação, 5) tradução dos sinais em comandos e 6) realimentação (MASON; BIRCH, 2003), conforme apresentado na Figura 6.

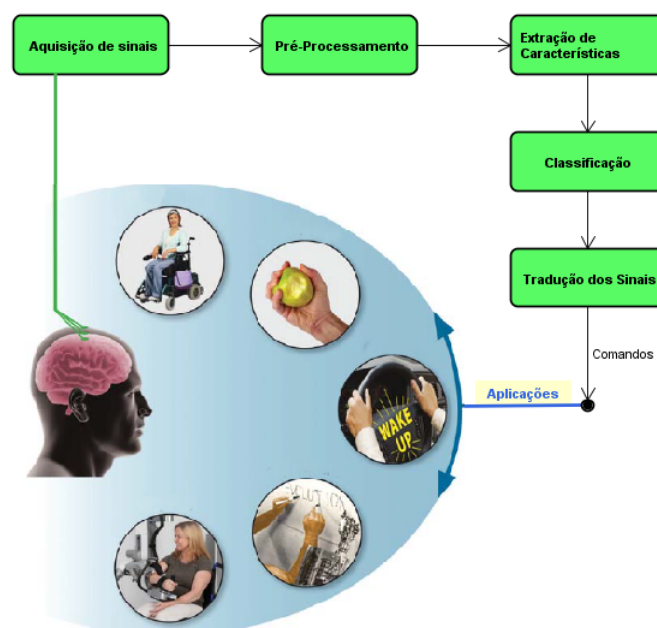


Figura 6 – Fluxograma de processos de uma BCI, onde pode ser aplicado em mecanismos motorizados, reabilitação muscular, alarmes, entre outros (Adaptado de (WOLPAW, 2012)).

1. **Medição de sinais:** Consiste na primeira etapa de um BCI, onde são medidos potenciais elétricos provenientes de atividades cerebrais (SIULY, 2012), os sinais utilizados neste trabalho foram adquiridos utilizando a EEG.
2. **Pré-Processamento:** Nesta etapa são utilizados filtros para limpar todo tipo de ruído do sinal e amplificando as informações relevantes dentro do sinal (SIULY, 2012).
3. **Extração de Características:** Nesta etapa são extraídos dos sinais valores relevantes, chamados de características (SIULY, 2012).
4. **Classificação:** Nesta etapa as características extraídas no processo anterior são rotuladas em determinadas classes (SIULY, 2012).
5. **Tradução em comandos:** Nesta etapa um comando é associado a cada uma das respectivas classes (SIULY, 2012).
6. **Realimentação:** Por fim é fornecido ao usuário da BCI uma informação a respeito do seu estado mental, qual atividade cerebral foi detectada (SIULY, 2012).

Um dos principais passos para a implementação de uma BCI é a **classificação**, pois é após este passo que é realizada a tradução dos sinais provenientes da EEG em comandos de controle (MASON; BIRCH, 2003). Para isso são utilizados algoritmos classificadores, onde um tipo de algoritmo de classificação é o LDA, que é detalhado na seção 2.5.

São utilizadas estratégias mentais para definir ao usuário tarefas mentais para gerar características padrões nos sinais cerebrais de acordo com determinada tarefa, para que o classificador possa interpretar corretamente (SIULY, 2012). Uma das mais comuns é imagética motora, que caracteriza a imaginação do movimento de um membro do corpo humano (SIULY, 2012).

## 2.4 BCI Competition

A *BCI Competition* é uma competição que promove o desenvolvimento e melhoria da tecnologia voltada para as BCIs, onde são submetidas diferentes técnicas de análise de dados cerebrais (BLANKERTZ et al., 2006). Já foram realizadas quatro edições da competição, nos anos de 2001, 2002, 2004 e 2008 (BLANKERTZ et al., 2006). Em cada uma destas competições são fornecidos publicamente sinais cerebrais, adquiridos em laboratórios especializados (BLANKERTZ et al., 2006). Estes sinais são divididos em dois conjuntos de dados, os dados de treinamento e os dados de teste, que são utilizados para treinamento e teste dos algoritmos dos participantes (BLANKERTZ et al., 2006).

### 2.4.1 *BCI Competition III*

O objetivo do *BCI Competition III* é validar as metodologias de classificação e processamento de sinais cerebrais aplicados em BCIs desenvolvidas pelos participantes da competição (BLANKERTZ et al., 2005). Esta edição foi realizada entre Maio e Junho de 2004, onde foram disponibilizados 8 *datasets* (I, II, IIIa, IIIb, IVa, IVb, IVc e V), desenvolvidos com a participação de 49 laboratórios especializados (BLANKERTZ et al., 2006). Para cada um dos *datasets* foram realizadas diferentes tarefas que estimulam atividades cerebrais durante a aquisição dos sinais, configurando assim um objetivo específico para cada um dos *datasets* (BLANKERTZ et al., 2005).

### 2.4.2 *BCI Competition III - Dataset IVa*

O *dataset IVa* refere-se a um conjunto de dados adquiridos através da EEG, onde os sujeitos (indivíduos nos quais foram capturados os sinais) foram submetidos a estimular o cérebro por imagética motora, através de indicações visuais (BLANKERTZ et al., 2006). Os indivíduos foram submetidos a realizarem três tarefas, indicadas visualmente por 3.5s cada tarefa, sendo interrompidas em períodos aleatórios entre 1.75s e 2.25s, onde o sujeito era submetido a um período de relaxamento (BLANKERTZ et al., 2006). As três tarefas de imagéticas motoras foram: (L) mão esquerda, (R) mão direita e (F) pé direito (BLANKERTZ et al., 2006).

Foram adquiridos sinais de 5 sujeitos rotulados em *aa*, *al*, *av*, *aw* e *ay*, onde foram executadas no total 280 tarefas por cada sujeito, algumas previamente rotulada (dados de treinamento) em cada instante de tempo onde a tarefa foi executada, outras não rotuladas (dados de teste) (BLANKERTZ et al., 2005). Estes sinais foram adquiridos, tratados e disponibilizados por *Fraunhofer FIRST*, *Intelligent Data Analysis Group* (Head: Klaus-Robert Müller), and *Charité University Medicine Berlin, Campus Benjamin Franklin, Department of Neurology, Neurophysics Group* (BLANKERTZ et al., 2006). A tabela 1 apresenta a quantidade de tarefas previamente classificadas (nomeados #tr) e a quantidade de tarefas não classificadas (nomeadas #te) para cada sujeito.

Tabela 1 – Número de tarefas rotuladas e não rotuladas por sujeito (BLANKERTZ et al., 2006).

Sujeitos	#tr	#te
<i>aa</i>	168	112
<i>al</i>	224	56
<i>av</i>	84	196
<i>aw</i>	56	224
<i>ay</i>	28	252

## 2.5 Linear Discriminant Analysis

O LDA é um classificador desenvolvido para explorar as informações no reconhecimento de padrões supervisionados, as informações conhecidas são contidas num vetor de treinamento previamente disponibilizado (IZENMAN, 2008). No algoritmo do LDA as informações de maior relevância são descobertas, enquanto que as de menor são eliminadas. O critério usado pelo algoritmo é obter as dimensões que possuem as características mais distintas das classes padrão. (A Expert System for Stomach Cancer Images with Artificial Neural Network by using HOG Features and Linear Discriminant Analysis: HOG LDA ANN )

Para um melhor entendimento, supõe-se a existência de um conjunto de dados  $\vec{x}$ , onde  $\vec{x} = (x_1, x_2, \dots, x_n)^T$  e  $T$  indicando transposição, com características multivariadas, e que cada dado seja conhecido devido ser proveniente de uma das classes  $y$ , tal que, são predefinidas com características semelhantes aos dados. As classes podem ser exemplificadas como sendo: espécies de plantas, presença ou ausência de uma condição médica específica, diferentes tipos de tumores, tipos de veículos automotores entre outros. Para separar as classes conhecidas uma das outras, é atribuído um rótulo a cada classe, então os dados são representados como dados rotulados (IZENMAN, 2008).

Devido a indispensabilidade de diminuir as dimensões dos dados de um determinado conjunto, o objetivo do LDA é reduzir a dimensão do espaço de conjunto de dados, resolvendo o inconveniente da sobreposição (SINGH; PRAKASH; CHANDRASEKARAN, 2016).

O LDA tem a proposta de encontrar uma transformação ótima para maximizar a proporção de acordo com a equação 2.1. com isso encontrando o vetor  $W$  que proporciona a melhor discriminação. (A Linear Discriminant Analysis Using Weighted Local Structure Information)

$$J(W) = \frac{W^T S_B W}{W^T S_W W} \quad (2.1)$$

Onde  $S_B$  é a matriz de dispersão entre as classes e  $S_W$  a matriz de dispersão dentro das classes.

$S_B$  é obtido segundo a equação 2.2

$$S_B = \sum_{j=1}^c n_j (m_j - \bar{m})(m_j - \bar{m})^T \quad (2.2)$$

onde  $c$  é o número de classes, e  $\bar{m}$  o vetor média e  $m_j$  é o vetor de média dos dados pertinentes à classe  $j$  e  $x$  é o vetor de treinamento.

$$\bar{m} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.3)$$



$$m = \frac{1}{n_j} \sum_{x \in c_j} x \quad (2.4)$$

A matriz de dispersão entre classes é definida pela equação 2.5,

$$S_B = \sum_{j=1}^c \sum_{x \in C_j} (X - m_j)(X - m_j)^T \quad (2.5)$$

Para  $S_W$  sendo uma matriz não-singular, a solução da equação 2.1, pode ser escrita como:

$$S_W^{-1} S_B W = \lambda W \quad (2.6)$$

onde  $\lambda$  é o auto-valor correspondente ao auto-vetor  $W$ .

## 2.6 System-on-Chip

*System-on-Chip* (SoC), implica que todo sistema que contém funcionalidades implementadas em hardware e software se encontra em um único chip de silício, combinando processamento, lógica de alta velocidade, interface, memória entre outros componentes ao invés de uma implementação maior em vários chips físicos diferentes agrupados em uma placa de circuito impresso (CROCKETT et al., 2014).

São vários os argumentos a favor da escolha de um SoC a uma placa de circuito impresso, pode-se citar que a solução é de menor custo, viabiliza transferência de dados mais rápidas e seguras entre vários elementos do sistema, possui maior velocidade geral do sistema, menor consumo de energia entre vários outros elementos que fortalecem a escolha de um SoC em sistemas discretos com componentes equivalentes (CROCKETT et al., 2014).

### 2.6.1 Arquitetura Simplificada de um SoC

O conjunto da arquitetura pode se dividir em dois sistemas, sistema de hardware e sistema de software.

No **Sistema de Hardware** encontram-se todos os periféricos, memórias e processadores, para conecta-los existe um barramento de comunicação responsável por isso. Já no **Sistema de Software**, o software é do tipo stack, e funciona sobre o processador, que também sustenta os aplicativos que geralmente têm um Sistema Operacional (SO) para gerência, uma camada em um nível mais baixo faz a interface com o sistema de hardware (CROCKETT et al., 2014).

Diagrama



### 2.6.2 Zybo-Board

A Zybo é uma plataforma de desenvolvimento (figura 7), que é equipada com o Z-7010 este que é o menor integrante da família Xilinx Zynq-7000. Baseado na arquitetura Xilinx<sup>®</sup> *All-Programmable System-on-Chip* (AP SoC) o Z-7010 possui em seu encapsulamento um processador ARM Cortex-A9 de dois núcleos e um Xilinx 7-Series (FPGA). Esse chip combinado com memórias, entradas e saídas de áudio e vídeo, USB, Ethernet, slot SD entre outros periféricos e suportes proporciona um *kit* para desenvolvedores que procuram uma plataforma de baixo custo sem perder a capacidade de processamento do Zynq AP SoC (DIGILENT, 2017).

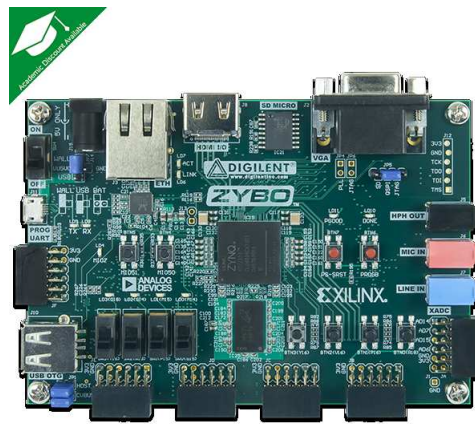


Figura 7 – Placa de Desenvolvimento Zybo-Board. Fonte: (DIGILENT, 2017)

## 2.7 Estado da Arte

A implementação de algoritmos de classificação em FPGA não é novidade. Em 2003 (YANG; PAINDAVOINE, 2003) descreveu um algoritmo de uma rede neural para reconhecimento de faces, obtendo uma acurácia de 92%. Neste mesmo trabalho realizou a comparação com a implementação em mais duas plataformas *Zero Instruction Set Computer* e em DSPs, onde obteve melhor resultado na implementação em DSP com acurácia de 98%.

Os algoritmos de classificação por padrão possuem estrutura bem definida, sobre o *BCI Competition - Data Set IVa*, os melhores resultados obtidos foram de (WANG et al., 2004), onde utilizando do classificador LDA, conseguiu acurácias de 94.17% geral, 95.5% sobre o sujeito (aa), 100.0% sobre o sujeito (al), 80.6% sobre o sujeito (av) 100.0% sobre o sujeito (aw) 97.6% sobre o sujeito (ay).

A tabela 2 apresenta a acurácia obtida por (LOTTE; GUAN, 2011) no ano de 2010, no desenvolvimento do algoritmo de classificação LDA, utilizando do algoritmo *Common*

*Spatial Pattern* (CSP) para maximizar a variância do filtro passa-faixa utilizados pelo EEG para uma das classes, enquanto minimiza a variância para as outras demais classes.

Tabela 2 – Acurácia de classificação do algoritmo LDA utilizando do algoritmo CSP para maximização de variância de classes. (LOTTE; GUAN, 2011)

Sujeitos	Acurácia do LDA com algoritmo CSP
<i>aa</i>	66,70%
<i>al</i>	96,43%
<i>av</i>	47,45%
<i>aw</i>	71,88%
<i>ay</i>	49,60%

## 3 Proposta Metodológica

Este capítulo apresenta os procedimentos e métodos a serem utilizados para desenvolvimento deste presente trabalho, detalhando as metodologias *Bottom-Up*, utilizadas nas implementações tanto em hardware quanto em software, além das linguagens de programação utilizadas nas implementações e suas respectivas ferramentas. Por fim é apresentado um cronograma de atividades.

### 3.1 Implementação em hardware

Esta seção apresenta os dispositivos e ferramentas a serem utilizados neste presente trabalho, além das metodologias a serem desenvolvidas, para implementação do algoritmo de treinamento do classificador LDA em Hardware.

#### 3.1.1 Dispositivos e ferramentas

Visando as grandes vantagens de utilização dos SoCs, o algoritmo de treinamento do classificador LDA será implementado nas lógicas digitais da FPGA na plataforma Zynq da placa *Zybo-Board*, utilizando os *IP-Cores*, que são blocos matemáticos desenvolvidos com propriedades intelectuais (MUNOZ et al., 2010) em unidade de ponto flutuante desenvolvidos por (MUNOZ et al., 2010) e a linguagem de programação *VHSIC Hardware Description Language* (VHDL), a fim de paralelizar os processos do algoritmo. Para o mapeamento do algoritmo em VHDL será utilizado a plataforma *Vivado HLS*, que possui todas as ferramentas necessárias para descrição, simulação, implementação e mapeamento hardware que descreve o algoritmo na FPGA (CROCKETT et al., 2014).

#### 3.1.2 Metodologias de desenvolvimento

Para desenvolvimento do algoritmo será adotada a metodologia *bottom-up*, onde cada sub-bloco desenvolvido é testado antes de ser inserido ao bloco principal, bloco de integração de todos sub-blocos do projeto, também conhecido como *Top module*. Após a implementação e simulação do *Top module*, será utilizado para teste e validação do hardware o *dataset IVa* do *BCI Competition III*, além de uma análise estatística do erro apresentado quando comparado com o desenvolvimento na plataforma *Matlab* por (LOTTE; GUAN, 2011).

Para validação da eficiência do SoC serão coletados os dados das seguintes características:

- Consumo de hardware: LUTs, FFs, blocos de DSP, blocos de memória RAM, I/O, MUX;
- Dados de desempenho: frequência de operação e tempo de execução;
- Estimação do consumo de potência.

Os processos de implementação do hardware em um SoC é apresentado na Figura

8.

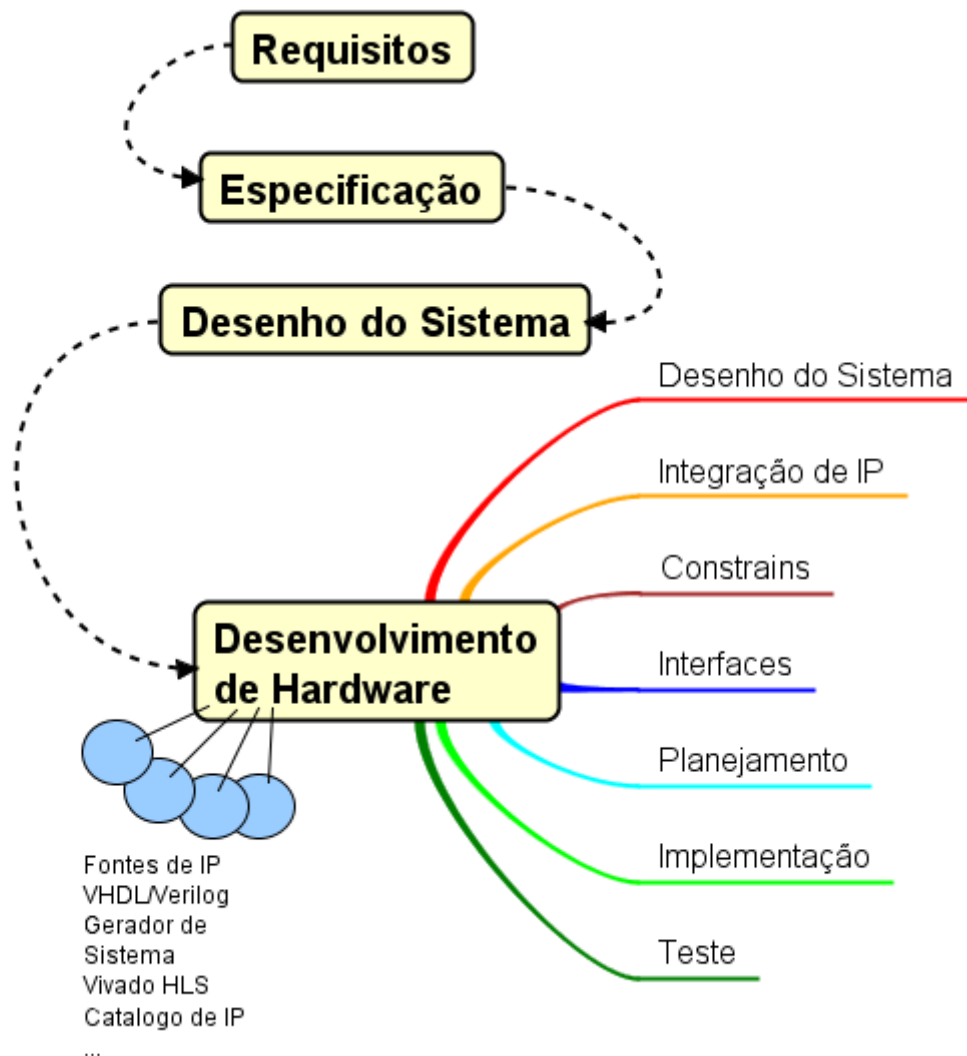


Figura 8 – Fluxograma de implementação de hardware em um SoC. Adaptado de (CROCKETT et al., 2014)

## 3.2 Implementação em software

São apresentados nesta seção, os materiais, métodos e técnicas para a implementação do algoritmo de treinamento do classificador LDA em software.

### 3.2.1 Materiais

Dispondo do SoC embarcado no kit de desenvolvimento da Digilent (*Zybo-Board*) , a implementação do algoritmo em software, explorará o processador *ARM<sup>®</sup> Cortex<sup>®</sup> -A9* de dois núcleos no *Xilinx Zynq<sup>®</sup> -7000*, usando também o auxílio do sistema operacional *xilinx*, a codificação do algoritmo será feita dipondo das linguagens de programação C e C++.

### 3.2.2 Métodos e Técnicas

Para o auxílio desta implemetação, será instalado o sistema operacional *Xilinx*, os recursos e passos necessários para a instalação do S.O estão contidos no ([CROCKETT et al., 2014](#)). O processo para implementação seguirá o método *bottom-up*, blocos de códigos menores serão implementados e testados separadamente, com a finalização de todos os blocos finalmente será integrado e testado por completo, as entradas para programa desenvolvido são provinientes do *dataset IVa* do *BCI Competition III*.

Os dados para análise pós implementação e teste serão:

- Consumo de memória
- Tempo de execução
- Consumo de potência
- Erro da implementação em comparação com os resultados obtidos por ([LOTTE; GUAN, 2011](#))

## 3.3 Data Set IVa

Em ambas as implementações serão utilizadas o *Data Set IVa* da *BCI Competition III* ([BLANKERTZ et al., 2006](#)), para efeito de testes e validações do sistema. Os dados *Data Set IVa* foram adquiridos e armazenados utilizando amplificadores do tipo *BrainAmp* e uma capa de eletrodos de 128 canais. Foram utilizados 118 canais de EEG posicionados de acordo com o sistema 10/20. Cada um destes canais foram filtrados em banda passante, utilizando um filtro *butterworth* de quinta ordem entre as frequências de 0.05 e 200 Hz, posteriormente foram digitalizados com uma frequência de amostragem de 1 kHz com precisão de 16 bits, apresentando uma resolução de 0.1  $\mu$ V, além disso também foram disponibilizados os mesmos dados com uma frequência de amostragem de 100 Hz ([BLANKERTZ et al., 2005](#)).

## 3.4 Cronograma de Atividades

Esta seção apresenta o cronograma de desenvolvimento deste presente trabalho.

Tabela 3 – Cronograma de atividades já desenvolvidas para este presente trabalho

<b>Atividade</b>	<b>Agosto</b>	<b>Setembro</b>	<b>Outubro</b>	<b>Novembro</b>	<b>Dezembro</b>
<i>Pesquisa Bibliográfica</i>	XO	XO	XO	XO	XO
<i>Escolha do Tema</i>	XO	XO			
<i>Pesquisa da Base de Dados</i>	XO	XO	XO		
<i>Pesquisa de Algoritmos já Desenvolvidos</i>		XO	XO	XO	
<i>Análise do Algoritmo Desenvolvido por (LOTTE; GUAN, 2011)</i>				XO	XO
<i>Reprodução dos Resultados obtidos por (LOTTE; GUAN, 2011)</i>				XO	XO
<i>Documentação do Projeto</i>			XO	XO	XO
<i>Apresentação do Projeto</i>					XO

As atividades já desenvolvidas são apresentadas no cronograma da tabela ?? As atividades a serem desenvolvidas por cada um dos autores é apresentada no cronograma da Tabela 4, onde as atividades atribuídas ao autor Heleno da Silva Moraes são referenciada simbolicamente com a letra "X" enquanto as atividades atribuídas ao autor Oziel da Silva Santos são referenciadas simbolicamente pela letra "O".

Tabela 4 – Cronograma de atividades a serem desenvolvidas para este presente trabalho.

<b>Atividade</b>	<b>Fevereiro</b>	<b>Março</b>	<b>Abril</b>	<b>Maiο</b>	<b>Junho</b>	<b>Julho</b>
<i>Pesquisa Bibliogrfica</i>	XO	XO	XO	XO	XO	XO
<i>Implementao do Algoritmo em FPGA</i>	X	X	X	X		
<i>Teste da Implementao em FPGA</i>	X	X	X	X	X	
<i>Validao da Implementao em FPGA</i>					X	X
<i>Levantamento das Caractersticas descritas na seo 3.1.2</i>					X	X
<i>Anlise Estatstica do Erro Obtido com a Implementao em FPGA</i>					X	X
<i>Instalao do SO Xilinx nos Core ARM</i>	O	O				
<i>Implementao do Algoritmo em Software Embarcado</i>		O	O	O		
<i>Testes do Algoritmo Implementado em Software Embarcado</i>	O	O	O	O	O	
<i>Validao da Implementao em Software Embarcado</i>					O	O
<i>Anlise Estatstica do Erro Obtido com a Implementao em Software Embarcado</i>					O	O
<i>Documentao do Projeto</i>	XO	XO	XO	XO	XO	XO
<i>Apresentao do Projeto</i>						XO





# Referências

- ALVAREZ, A.; LEMOS, I. d. C. Os neurobiomecanismos do aprender: a aplicação de novos conceitos no dia-a-dia escolar e terapêutico. *Revista Psicopedagogia*, Associação Brasileira de Psicopedagogia, v. 23, n. 71, p. 181–190, 2006. Citado 4 vezes nas páginas 15, 29, 30 e 31.
- BLANKERTZ, B. et al. *BCI Competition III*. 2005. Disponível em: <[http://www.bbc.de/competition/iii/desc\\_IVa.html](http://www.bbc.de/competition/iii/desc_IVa.html)>. Citado 2 vezes nas páginas 36 e 43.
- BLANKERTZ, B. et al. The bci competition iii: validating alternative approaches to actual bci problems. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 14, n. 2, p. 153–159, June 2006. ISSN 1534-4320. Citado 5 vezes nas páginas 17, 27, 35, 36 e 43.
- BRUNELLI, R. *Template matching techniques in computer vision: theory and practice*. [S.l.]: John Wiley & Sons, 2009. Citado na página 25.
- CAMPISI, P.; ROCCA, D. L.; SCARANO, G. Eeg for automatic person recognition. *Computer*, IEEE, v. 45, n. 7, p. 87–89, 2012. Citado 3 vezes nas páginas 15, 32 e 33.
- CLARK NASHAAT BOUTROS, M. M. D. *The Brain and Behavior: An Introduction to Behavioral Neuroanatomy*. 2. ed. Cambridge University Press, 2005. ISBN 0521840503,9780521840507. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=D00D7BF8931D1C6313FA52A331272806>>. Citado na página 29.
- CROCKETT, L. H. et al. *The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable Soc*. [S.l.]: Strathclyde Academic Media, 2014. Citado 7 vezes nas páginas 15, 26, 27, 38, 41, 42 e 43.
- DIGILENT. *Zybo Board Reference Manual*. 2017. Disponível em: <<https://reference.digilentinc.com/reference/programmable-logic/zybo/reference-manual>>. Citado 2 vezes nas páginas 15 e 39.
- GUGER, C. et al. *Brain-Computer Interface Research: A State-of-the-Art Summary*. 1. ed. Springer-Verlag Berlin Heidelberg, 2013. (SpringerBriefs in Electrical and Computer Engineering). ISBN 978-3-642-36082-4,978-3-642-36083-1. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=8EF0FF02FCA45F572B14D32FAFE8B2AD>>. Citado 2 vezes nas páginas 33 e 34.
- IZENMAN, A. J. *Modern multivariate statistical techniques*. [S.l.]: Springer, 2008. v. 1. Citado na página 37.
- KANDEL, e. a. E. R. *Principles of Neural Science*. 5. ed. The address: Mc Graw Hill, 2013. v. 2. An optional note. ISBN 978007181001-2. Citado 3 vezes nas páginas 25, 29 e 30.
- LOTTE, F. *Study of electroencephalographic signal processing and classification techniques towards the use of brain-computer interfaces in virtual reality applications*. Tese (Doutorado) — INSA de Rennes, 2008. Citado 2 vezes nas páginas 25 e 26.

- LOTTE, F.; GUAN, C. Regularizing common spatial patterns to improve bci designs: Unified theory and new algorithms. *IEEE Transactions on Biomedical Engineering*, v. 58, n. 2, p. 355–362, Feb 2011. ISSN 0018-9294. Citado 10 vezes nas páginas 17, 25, 26, 27, 33, 39, 40, 41, 43 e 44.
- MASON, S. G.; BIRCH, G. E. A general framework for brain-computer interface design. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 11, n. 1, p. 70–85, March 2003. ISSN 1534-4320. Citado 4 vezes nas páginas 25, 27, 34 e 35.
- MUÑOZ, D. M. et al. Tradeoff of fpga design of a floating-point library for arithmetic operators. *Journal of Integrated Circuits and Systems*, v. 5, n. 1, p. 42–52, 2010. Citado na página 41.
- RAO, R. P. *Brain-computer interfacing: an introduction*. [S.l.]: Cambridge University Press, 2013. Citado 2 vezes nas páginas 31 e 32.
- SHASHOA, N. A. A. et al. Classification depend on linear discriminant analysis using desired outputs. In: IEEE. *Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2016 17th International Conference on*. [S.l.], 2016. p. 328–332. Citado na página 26.
- SINGH, A.; PRAKASH, B. S.; CHANDRASEKARAN, K. A comparison of linear discriminant analysis and ridge classifier on twitter data. In: IEEE. *Computing, Communication and Automation (ICCCA), 2016 International Conference on*. [S.l.], 2016. p. 133–138. Citado na página 37.
- SIULY, S. *Analysis and Classification of EEG Signals*. Dissertação (Mestrado) — University of Southern Queensland, <https://www.springer.com/gp/book/9783319476520>, 7 2012. Citado 7 vezes nas páginas 15, 25, 30, 31, 32, 33 e 35.
- SIULY, S.; LI, Y.; ZHANG, Y. *EEG Signal Analysis and Classification: Techniques and Applications*. [S.l.]: Springer, 2017. Citado 4 vezes nas páginas 25, 26, 31 e 32.
- TEPLAN, M. et al. Fundamentals of eeg measurement. *Measurement science review*, v. 2, n. 2, p. 1–11, 2002. Citado na página 30.
- THEODORIDIS, S.; KOUTROUMBAS, K. et al. *Pattern recognition*. [S.l.]: Academic press London, 1999. Citado na página 26.
- WANG, Y. et al. Bci competition 2003-data set iv:an algorithm based on cssd and fda for classifying single-trial eeg. *IEEE Transactions on Biomedical Engineering*, v. 51, n. 6, p. 1081–1086, June 2004. ISSN 0018-9294. Citado na página 39.
- WOLPAW, E. W. W. J. R. *Brain-Computer Interfaces: Principles and Practice*. Oxford University Press, 2012. ISBN 9780195388855 0195388852. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=46D2894771F9769A0D5B429B681CBF33>>. Citado 3 vezes nas páginas 15, 33 e 34.
- YANG, F.; PAINDAVOINE, M. Implementation of an rbf neural network on embedded systems: real-time face tracking and identity verification. *IEEE Transactions on Neural Networks*, v. 14, n. 5, p. 1162–1175, Sept 2003. ISSN 1045-9227. Citado na página 39.