

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**DEPARTAMENTO ACADÊMICO DE INFORMÁTICA**

**CURSO DE ENGENHARIA DE COMPUTAÇÃO**

**RUANI LAZZAROTTO**

**SISTEMA DE RECONHECIMENTO DE PADRÕES DO ALFABETO DA  
LÍNGUA BRASILEIRA DE SINAIS UTILIZANDO MICROCONTROLADOR**

**TRABALHO DE CONCLUSÃO DE CURSO 2**

**PATO BRANCO**

**2016**

**RUANI LAZZAROTTO**

**SISTEMA DE RECONHECIMENTO DE PADRÕES DO ALFABETO DA  
LÍNGUA BRASILEIRA DE SINAIS UTILIZANDO  
MICROCONTROLADOR**

Trabalho de Conclusão de Curso de graduação apresentado à disciplina de Trabalho de Conclusão de Curso 2, do Curso de Engenharia de Computação da Universidade Tecnológica Federal do Paraná – UTFPR, Campus Pato Branco, como requisito parcial para obtenção do título de Bacharel.

Orientador: Prof. Dr. Fábio Luiz Bertotti

**PATO BRANCO**

**2016**



---

## TERMO DE APROVAÇÃO

Às 16 horas do dia 01 de julho de 2016, na sala V006, da Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco, reuniu-se a banca examinadora composta pelos professores Fabio Luiz Bertotti (orientador), Dalcimar Casanova e Kathya Silvia Collazos Linares para avaliar o trabalho de conclusão de curso com o título **Sistema de reconhecimento de padrões do alfabeto da língua brasileira de sinais utilizando microcontrolador**, do aluno **Ruani Lazzarotto**, matrícula 01115073, do curso de Engenharia de Computação. Após a apresentação o candidato foi arguido pela banca examinadora. Em seguida foi realizada a deliberação pela banca examinadora que considerou o trabalho aprovado.

---

Fabio Luiz Bertotti  
Orientador (UTFPR)

---

Dalcimar Casanova  
(UTFPR)

---

Kathya Silvia Collazos Linares  
(UTFPR)

---

Beatriz Terezinha Borsoi  
Coordenador de TCC

---

Pablo Gauterio Cavalcanti  
Coordenador do Curso de  
Engenharia de Computação

A Folha de Aprovação assinada encontra-se na Coordenação do Curso.

## **AGRADECIMENTOS**

Agradeço aos meus pais Omero Lazzarotto e Ana Paulina Zembrani Lazzarotto pela dedicação e suporte ao longo da vida e dos anos para concluir a graduação.

Agradeço à minha avó Tervina Ecker por me acolher em sua casa e ter me tratado tão gentilmente ao longo dos últimos anos.

Agradeço à Marília Ponciano por ter me servido de inspiração para o tema do trabalho e por ter me motivado durante os períodos difíceis.

Ao meu orientador Fábio Luiz Bertotti por se interessar pela proposta deste trabalho, por ter prestado fundamental ajuda ao longo do desenvolvimento, sempre se dispondo a resolver os problemas mais críticos de projeto. Agradeço profundamente a se dispor em me guiar e apoiar em todas as etapas que me levaram a concluir o presente trabalho.

A todos os amigos, colegas e professores que me ajudaram a superar, de alguma maneira, as dificuldades encontradas durante o curso.

## RESUMO

LAZZAROTTO, Ruani. Sistema de Reconhecimento de Padrões do Alfabeto da Língua Brasileira de Sinais utilizando Microcontrolador. 2016. 103 f. Monografia (Trabalho de Conclusão de Curso 2) – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná, 2016.

A Língua Brasileira de Sinais ou Libras é a língua utilizada por grande parte da população surda do país. Atualmente, observa-se a busca em tornar a sociedade mais inclusiva para pessoas que possuam algum tipo de deficiência e uma proposta de inclusão é o incentivo ao aprendizado da Libras para que a comunicação entre surdos e ouvintes possa ser possível sem o uso de intérpretes. O presente trabalho tem como objetivo o desenvolvimento de um sistema, utilizando luva sensora, para o reconhecimento dos padrões dos sinais da Libras que constituem as letra do alfabeto. Uma técnica simplificada de sensoriamento da luva é empregada utilizando sensores indutivos de fabricação própria, além do desenvolvimento do circuito de condicionamento dos sinais de resposta dos sensores. Os dados são processados utilizando uma placa de desenvolvimento de sistemas microcontrolados onde é implementado um algoritmo de reconhecimento de padrões baseado em redes neurais artificiais. As letras reconhecidas pelo sistema são apresentadas em um *display* digital. Apesar das limitações de informação disponível do sensoriamento modelado, os sensores desenvolvidos apresentaram significativo potencial na determinação de configurações de mão. Os resultados obtidos com o reconhecimento dos padrões mostraram que o sistema, dentro do escopo de modelagem, apresentou viabilidade para ser utilizado em aplicações que possam desenvolver as habilidades em datilografia de palavras para alunos iniciantes em Libras.

**Palavras-chave:** Língua Brasileira de Sinais. Luva Sensora. Microcontrolador. Sensor Indutivo. Reconhecimento de Padrões.

## ABSTRACT

LAZZAROTTO, Ruani. Pattern Recognition System for Brazilian Sign Language's Alphabet using Microcontroller. 2016. 103 f. Monografia (Trabalho de Conclusão de Curso 2) – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná, 2016.

The Brazilian Sign Language is the language used by most of the deaf people in Brazil. Currently, there is a challenge to make the society more inclusive for people who have some kind of disability, and a proposal for inclusion is encouraging the learning of Sign Languages for communication between deaf and hearing people may be possible without the use of interpreters. This work aims to develop a system using sensor glove, for the recognition of patterns of the Brazilian Sign Language signs constituting the letter of the alphabet. A simplified technique for project sensor glove is employed using handmade inductive sensors, and the development of the electronic circuit for sensor response signal conditioning. The data is processed using a microcontroller systems development board using an pattern recognition algorithm based on artificial neural network. The letters recognized by the system are displayed on a digital display. Despite the limitations of information available from the sensor glove, the developed sensors showed significant potential in determining hand shapes. The results obtained with the pattern recognition system showed the viability for use the system in applications for development skills in dactylogy for beginners in Brazilian Sign Language

**Keywords:** Brazilian Sign Language. Microcontroller. Inductive sensor. Pattern Recognition.

## LISTA DE FIGURAS

Figura 1 - As 46 configurações de mão da Libras. ....	24
Figura 2 - Os sinais Telefone e Branco e suas configurações de mão. ....	24
Figura 3 - Os sinais de “aprender” e “sábado” e seus pontos de articulação ....	25
Figura 4 - Sinais de “verde” e “gelado” e seus movimentos ....	26
Figura 5 - Alfabeto Manual. ....	27
Figura 6 - As palavras “Mateus”, “Maria”, e “ <i>chip</i> ” em datilologia. ....	28
Figura 7 - Elementos de um sistema de processamento de imagens ....	30
Figura 8 - CyberGlove com 18 sensores. ....	32
Figura 9 - Esquema elétrico dos terminais do sensor de contato mecânico. ....	33
Figura 10 - Funcionamento do Flex Sensor. ....	34
Figura 11 - Exemplo de posicionamento das bobinas. ....	36
Figura 12 - Indução mútua ....	37
Figura 13 - Multiplexagem. ....	40
Figura 14 - AmpOp básico. ....	41
Figura 15 - Seguidor unitário. ....	42
Figura 16 - Retificador onda completa de dois estágios. ....	43
Figura 17 - Componentes essenciais de um microcontrolador. ....	45
Figura 18 - Diagrama de um sistema de reconhecimento de padrões. ....	49
Figura 19 - Modelo de um neurônio. ....	51
Figura 20- Exemplo de rede <i>feedforward</i> de camada múltipla. ....	52
Figura 21 - Luva MaxiFlex®. ....	56
Figura 22 - Módulo LCD. ....	57
Figura 23 - Placa de desenvolvimento MSP430F5529. ....	57
Figura 24 - Diagrama em blocos dos circuitos de controle da luva sensora. ....	62
Figura 25 - Diagrama de blocos do sistema. ....	65
Figura 26 - Topologia dos circuitos de sensoriamento. ....	67
Figura 27 - Conjunto modelado de sensores e geradores da luva. ....	68
Figura 28 - Sensor indutivo desenvolvido ....	71
Figura 29 - Diagrama do circuito de condicionamento de sinais. ....	72

Figura 30 - Circuito de alimentação dos geradores.....	73
Figura 31 - Circuito divisor de tensão.....	74
Figura 32 - Circuito de multiplexação dos sensores.....	74
Figura 33 - Circuito de retificação.....	75
Figura 34 - Circuito pós-retificação.....	76
Figura 35 - Circuito de regulação de tensão.....	77
Figura 36 - Fluxograma principal do sistema.....	77
Figura 37 - Circuito de condicionamento, microcontrolador e <i>display</i> . ....	86
Figura 38 - Vista frontal e lateral da luva sensora desenvolvida. ....	87
Figura 39 - Sistema completo e conectado. ....	87

## LISTA DE QUADROS

Quadro 1 - Metodologia de desenvolvimento de um sistema embarcado.....	44
Quadro 2 - As ferramentas de uma IDE.....	47
Quadro 3 - Descrição da localização dos geradores.....	68
Quadro 4 - Descrição da localização dos sensores.....	68
Quadro 5 - Mapeamento dos sensores para configuração da letra A.....	70
Quadro 6 - Características físicas e elétricas dos circuitos.....	70
Quadro 7 - Pares de ativação.....	79
Quadro 8 - Topologia de redes candidatas.....	80

## LISTA DE CÓDIGOS

Listagem 1 - Roteiro de criação e treinamento da rede neural.....	81
Listagem 2 - Rotina de varredura dos sensores.....	83
Listagem 3 - Representação dos neurônios.....	84
Listagem 4 - Rotina RNA.....	84
Listagem 5 - Função de pós processamento.....	85
Listagem 6 - Códigos das funções de seleção de canais.....	102
Listagem 7 - Código da função de normalização. ....	103

## LISTA DE TABELAS

Tabela 1 - Resposta do sensor S1 em relação à distância do gerador G1. ....	88
Tabela 2 - Resposta do sensor S6 em relação ao ângulo entre o gerador G4. ....	88
Tabela 3 - Resultado dos treinamentos das topologias candidatas. ....	90
Tabela 4 - Matriz de confusão. ....	91
Tabela 5 - Padrão de dados da letra A. ....	92
Tabela 6 - Comparação entre as respostas das redes simulada e implementada. ....	93

## LISTA DE SIGLAS

ADC	Analogic Digital Converter (Conversor Analógico/Digital)
ASL	American Sign Language (Língua Americana de Sinais)
CI	Circuito Integrado
CC	Corrente Contínua
CM	Configuração de Mão
CPU	Central Processing Unit (Unidade Central de Processamento)
LS	Línguas de Sinais
MCU	MicroController Unit Microcontrolador
PCI	Placa de Circuito Impresso
PWM	Pulse Width Modulation (Modulação por Largura de Pulso)
RNA	Rede Neural Artificial

## LISTA DE ACRÔNIMOS

FEM	Força Eletromotriz
INES	Instituto Nacional de Educação de Surdos
LIBRAS	Língua Brasileira de Sinais
MEM	Micro Eletromecânico

## SUMÁRIO

1	INTRODUÇÃO	16
1.1	CONSIDERAÇÕES INICIAIS	16
1.2	OBJETIVOS	18
1.2.1	Objetivo Geral	19
1.2.2	Objetivos Específicos	19
1.3	JUSTIFICATIVA	19
1.4	ORGANIZAÇÃO DO TRABALHO	20
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	LÍNGUA BRASILEIRA DE SINAIS	21
2.1.1	Histórico	21
2.1.2	Aspectos Estruturais da Libras	23
2.1.3	O Alfabeto Manual e Datilologia	26
2.1.4	A Datilologia e o Aprendizado da Libras	28
2.2	MÉTODOS AUTOMATIZADOS PARA RECONHECIMENTO DE LÍNGUA DE SINAIS	29
2.2.1	Método Baseado em Visão Computacional	29
2.2.2	Método Baseado em Luva Sensora	31
2.3	SENSORES	33
2.3.1	Sensor de Contato Mecânico	33
2.3.2	Sensor Resistivo de Flexão	34
2.3.3	Sensores de Aceleração	35
2.3.4	Sensores Indutivos	35
2.4	CONDICIONAMENTO DE SINAIS	38
2.4.1	Filtragem	39
2.4.2	Multiplexagem	39
2.4.3	Retificação	40
2.4.4	Amplificação e Isolação	40
2.4.5	Amplificador Operacional	41
2.4.5.1	Topologia de Retificador com AmpOps	42
2.5	SISTEMAS EMBARCADOS	43
2.5.1	Microcontrolador	44
2.5.2	Periféricos e Interfaces	46

2.5.3	Desenvolvimento de <i>Firmware</i> para Microcontroladores.....	46
2.6	RECONHECIMENTO DE PADRÕES.....	48
2.6.1	Sistemas de Reconhecimento de Padrões.....	48
2.6.2	Redes Neurais Artificiais.....	50
2.6.3	Arquiteturas de Redes Neurais.....	51
2.6.4	Aprendizagem e Algoritmo de Retro Propagação.....	53
2.7	ESTADO DA ARTE.....	53
3	MATERIAIS E MÉTODOS.....	55
3.1	MATERIAIS.....	55
3.1.1	Luva de Fixação dos Sensores.....	55
3.1.2	Material para Confeção dos Sensores e Geradores.....	56
3.1.3	<i>Display</i> .....	56
3.1.4	Placa de Desenvolvimento de Sistemas Microcontrolados.....	57
3.1.5	Ferramenta de Desenvolvimento de Código para o Microcontrolador.....	58
3.1.6	Ferramenta de Desenvolvimento de Redes Neurais.....	58
3.1.7	Ferramenta para Desenho Esquemático e Layout de Circuito Impresso.....	59
3.1.8	Bibliotecas e Exemplos de Códigos.....	59
3.2	METODOLOGIA.....	59
3.2.1	Desenvolvimento do Sensoriamento da Luva.....	60
3.2.2	Desenvolvimento de Circuitos de Condicionamento de Sinais Analógicos.....	61
3.2.3	Desenvolvimento do Código do Sistema Microcontrolado.....	62
3.2.4	Método para Reconhecimento de Padrões.....	63
3.2.5	Validação e Teste do Sistema.....	64
4	DESENVOLVIMENTO.....	65
4.1	VISÃO GERAL DO SISTEMA.....	65
4.2	SENSORIAMENTO DA LUVA.....	66
4.2.1	Modelagem do Sensoriamento.....	66
4.2.2	Dimensionamento das Bobinas.....	70
4.3	CIRCUITO DE CONDICIONAMENTO DE SINAIS.....	71
4.4	MODELAGEM DO <i>FIRMWARE</i> .....	77
4.4.1	Inicialização das Portas e Periféricos.....	78
4.4.2	Varredura dos Sensores e Geradores.....	79
4.4.3	Reconhecimento de Padrões.....	79
4.4.4	Simulação da Rede Neural.....	80

4.5 IMPLEMENTAÇÃO .....	82
4.5.1 Aquisição dos Dados da Luva .....	82
4.5.2 Implementação da Rede Neural .....	83
4.5.3 Interpretação da Saída da Rede Neural .....	85
5 RESULTADOS E DISCUSSÕES .....	86
5.1 ASPECTO FÍSICO DO SISTEMA.....	86
5.2 RESPOSTA DOS SENSORES.....	88
5.3 POSICIONAMENTO DOS SENSORES NA LUVA .....	89
5.4 TESTE DA CLASSIFICAÇÃO.....	90
6 CONSIDERAÇÕES FINAIS .....	94
6.1 TRABALHOS FUTUROS .....	95
APÊNDICE A.....	102

## 1 INTRODUÇÃO

Neste Capítulo será realizada uma breve contextualização sobre o tema tratado neste trabalho. Também serão apresentados os objetivos que se pretende alcançar ao longo do desenvolvimento do sistema. O Capítulo apresenta ainda a justificativa em realizar o sistema que será descrito. Por fim é apresentada como estão organizados os demais capítulos que compõem este trabalho.

### 1.1 CONSIDERAÇÕES INICIAIS

A língua de sinais é o principal modo de comunicação entre as pessoas com deficiência auditiva, uma vez que estas não conseguem perceber ou, em certos casos, gerar a informação acústica, é exigido, portanto, que a troca de informações seja feita de forma visual. No entanto, os portadores desta deficiência específica ainda possuem dificuldade de comunicação com as pessoas que ouvem normalmente, pois estes últimos em sua maioria não entendem os sistemas de linguagem de sinais, tais como a Língua Brasileira de Sinais.

A Língua Brasileira de Sinais ou Libras, instituída no Brasil como língua oficial, abrange as necessidades das pessoas com deficiência auditiva e está cada dia mais presente em ambientes inclusivos, em programas de televisão, eventos educacionais, em disciplinas acadêmicas, dentre outros espaços sociais (CARVALHO, 2011).

Segundo dados divulgados pelo Instituto Brasileiro de Geografia e Estatística (2010), a porcentagem da população que é portadora de alguma necessidade específica é de aproximadamente 24%, sendo que destes mais de nove milhões de pessoas possuem algum nível de deficiência auditiva. Para que essa parte da população possa ter uma plena vivência é necessário que cada vez mais se pense em soluções que possibilitem a inclusão dessas pessoas na sociedade.

Apesar dos esforços de inclusão da Libras na sociedade, ainda há muitos obstáculos a serem superados. A dificuldade de entendimento da língua por pessoas não familiarizadas com esta linguagem afeta negativamente a vida e os

relacionamentos entre as pessoas com e sem deficiência auditiva. Os deficientes auditivos normalmente se comunicam com ouvintes, quer através de intérpretes ou de escrita de texto. Embora os intérpretes possam facilitar a comunicação entre pessoas portadoras de deficiência auditiva e pessoas ouvintes, o custo deste serviço não é viável para a maioria dos deficientes, e seu envolvimento leva a uma perda de independência e privacidade. A comunicação de forma escrita também é bastante utilizada, porém torna-se inconveniente durante uma caminhada ou quando não se tenha papel e caneta disponível no momento (OZ, 2011).

Para que a sociedade seja mais inclusiva é necessário que tanto a pessoa com deficiência auditiva tenha condições de se adaptar aos diversos ambientes, seja aprendendo leitura labial e Português escrito, quanto à adaptação das pessoas ouvintes para interagir com as pessoas que possuam necessidades específicas através do entendimento da língua de sinais, por exemplo. Uma das maneiras de interessar as pessoas ouvintes a conhecerem e se interessarem em aprender a Libras é utilizando de aplicações de dispositivos móveis ou ainda em tecnologias de realidade virtual, onde o movimento do corpo, muito presente na Libras, pode ser usado como dispositivo de entrada de dados em um ambiente de ensino virtual das línguas de sinais.

O reconhecimento de gestos tem sido uma área ativa de investigação durante as últimas décadas. Além da busca por uma interação mais natural entre humanos e computadores, existem muitas aplicações interessantes na área da robótica, realidade virtual, tele manipulação, tele presença e na engenharia biomédica. Por exemplo, uma das aplicações na área de engenharia biomédica inclui o monitoramento do avanço de doenças degenerativas através da análise da capacidade de movimentação da mão, ou ainda para estimar com certa precisão a melhora do movimento adquirido durante sessões de fisioterapia (REBOLLAR; LINDEMAN; KYRIAKOPOULOS, 2002).

Muitos pesquisadores têm trabalhado para o reconhecimento de configurações de mão que compõem as diversas línguas de sinais, mas esta pesquisa apresenta grande dificuldade devido à complexidade do movimento da mão e do corpo em linguagens de sinais. Pesquisas de reconhecimento de língua de sinais podem ser classificadas em três categorias principais: (i) com base em visão computacional, (ii) com base em luva sensora e (iii) uma combinação de estes dois métodos (OZ; LEU, 2011, p. 1204).

O reconhecimento de sinais baseado em visão computacional depende de técnicas de processamento de imagens e extração de características para capturar e classificar os movimentos do corpo e a posição das mãos quando uma pessoa sinalizadora de Libras faz um sinal de uma determinada linguagem. A complexidade envolvida neste método pode ser reduzida através do uso de uma luva sensora ou (luva de dados) que permite o reconhecimento de movimentos e detecção de gestos das mãos. A combinação destes métodos permite que a dificuldade de reconhecimento de gestos mais complexos pela visão computacional, como o movimento dos dedos, possa ser compensada com o uso da luva sensora (OZ; LEU, 2011).

O trabalho tem como objetivo descrever como foi realizada a construção de um sistema de reconhecimento de configurações de mão que correspondam ao alfabeto manual da Libras. A pessoa sinalizadora, utilizando a luva sensora desenvolvida, poderá articular os sinais enquanto o sistema realiza a leitura dos sensores. O sistema então utilizará de uma técnica de reconhecimento de padrões utilizando de redes neurais e exibe a letra correspondente no alfabeto da Língua Portuguesa em um *display*.

Esse sistema possibilitará que estudantes de Libras possam conhecer o alfabeto da Libras e aprimorar suas habilidades de datilografia (soletração em Língua de Sinais). Uma das aplicações possíveis do sistema poderá ser, por exemplo, um jogo de soletração manual onde o usuário da luva deverá datilografar as palavras que surgirão no *display*. O sistema poderia apresentar uma pontuação de acordo com a taxa de acerto das palavras datilogradas.

## 1.2 OBJETIVOS

Essa subseção é dedicada à apresentação dos objetivos do trabalho que serão divididos entre um objetivo geral, enfatizando o resultado final que o trabalho pretende apresentar e objetivos específicos, resultados parciais que podem ser esperados para a realização do objetivo geral.

### 1.2.1 Objetivo Geral

Desenvolver um dispositivo do tipo luva sensora capaz de reconhecer configurações de mão, correspondente ao alfabeto manual da Língua Brasileira de Sinais (Libras), e realizar a tradução destas para o seu significado na Língua Portuguesa apresentando o resultado na forma de texto em um *display*.

### 1.2.2 Objetivos Específicos

- Pesquisa sobre as técnicas e sensores para aquisição da posição dos dedos;
- Desenvolver um sistema de aquisição e processamento de dados para os sensores escolhidos para esta aplicação;
- Aplicar um algoritmo para a classificação das configurações de mão, permitindo reconhecer as letras do alfabeto em Libras;
- Desenvolver uma interface do sistema utilizando um *display*;
- Teste e avaliação do sistema proposto.

## 1.3 JUSTIFICATIVA

Segundo Rosa (2009, p. 9) a inclusão das pessoas que apresentem necessidades específicas vai além do convívio ou a permanência no mesmo ambiente que pessoas ditas normais: “representa possibilidades de se rever as concepções e paradigmas, num profundo respeito pelas suas diferenças”. Portanto, a realização deste trabalho se justifica em função da necessidade em romper barreiras de comunicação entre pessoas portadoras de deficiência auditivas que se comunicam apenas pela Língua Brasileira de Sinais e pessoas ouvintes que não possuam conhecimento desta língua, além do estímulo em pesquisas de diferentes tecnologias que podem ser aplicadas para acessibilidade de pessoas que possuam algum tipo de necessidade específica.

O sistema de luva sensora por si só apresenta inúmeras aplicações em diversas áreas. De acordo com Fahn e Sun (2005) esse tipo de dispositivo tem estado presente em aplicações na área de realidade virtual, medicina, robótica e entre outras áreas onde o movimento do corpo pode ser usado como dado de entrada.

Apesar de o sistema proposto reconhecer apenas as configurações de mão que constituem o alfabeto da Libras, seu uso pode contribuir significativamente para o aprendizado do alfabeto desta língua por ouvintes ou deficientes auditivos em instituições de ensino ou ainda para melhorar a prática da datilologia de palavras em Libras.

#### 1.4 ORGANIZAÇÃO DO TRABALHO

O trabalho é composto de cinco capítulos. No Capítulo 1 é apresentada uma introdução do trabalho contendo uma breve contextualização, assim como os objetivos e a justificativa da realização do trabalho. No segundo Capítulo é apresentada uma revisão bibliográfica dos assuntos abordados no trabalho. Esse capítulo contém um breve histórico e características da Língua Brasileira de Sinais além dos aspectos relacionados ao alfabeto manual. O terceiro Capítulo é destinado aos materiais e metodologias que serão empregados na realização deste trabalho. O desenvolvimento do sistema é descrito no Capítulo quatro. Os resultados preliminares são apresentados no Capítulo cinco. Por fim, no Capítulo seis situam-se as considerações finais acerca da realização deste trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

Este Capítulo tem por objetivo realizar uma breve contextualização sobre os temas que serão utilizados no desenvolvimento deste projeto. Primeiramente, é apresentada uma abordagem sobre a Língua Brasileira de Sinais seguindo pelos métodos tipicamente utilizados para o reconhecimento de línguas de sinais. O capítulo segue com uma breve descrição sobre os sensores que geralmente utilizados em sistemas de luva sensora. As características de sistemas embarcados e métodos de reconhecimento de padrões também são introduzidos. Por fim é apresentado o estado da arte.

### 2.1 LÍNGUA BRASILEIRA DE SINAIS

As Línguas de Sinais (LS) são línguas que surgem naturalmente nas comunidades surdas e não se tratam de mímicas e gestos icônicos, utilizados como tentativa de comunicação, mas de línguas com estruturas gramaticais próprias (GESSER, 1971). Atribui-se às Línguas de Sinais o status de língua, pelo fato de que elas também apresentam os níveis linguísticos das línguas orais. O que apenas diferencia as Línguas de Sinais das demais línguas é a sua modalidade visual-espacial (GESSER, 1971).

Segundo Gesser (1971), “a Língua Brasileira de Sinais, ou Libras, é uma Língua comparável em complexidade e expressividade a qualquer língua oral capaz de expressar ideias sutis, complexas e abstratas”. Assim como em cada país as pessoas ouvintes falam diferentes línguas, também as pessoas surdas também possuem suas próprias línguas, existindo assim Língua de Sinais Francesa, Língua de Sinais Alemã, e entre outras (GESSER, 1971).

#### 2.1.1 Histórico

A primeira iniciativa de incentivo à educação de pessoas surdas no Brasil foi realizada quando o imperador D. Pedro II trouxe Eduard Huet, um professor surdo francês com mestrado em Paris com o intuito de fundar um instituto para a

educação de surdos. Em 26 de setembro de 1857, surge no Rio de Janeiro a primeira escola para surdos no Brasil, o Imperial Instituto de Surdos-Mudos, intitulada atualmente de Instituto Nacional de Educação dos Surdos ou INES. Também nesse dia comemora-se o Dia Nacional dos Surdos no Brasil (ALMEIDA; ALMEIDA, 2012).

O livro intitulado “Iconographia dos Signaes dos Surdos-Mudos”, de autoria do aluno José da Gama Flausino, vinculado ao INES e publicado em 1873, é, segundo Ramos (2003), a publicação de maior relevância no sentido histórico sobre a Língua Brasileira de Sinais, com ilustrações de sinais separados por categorias (animais, objetos, etc.). Esse livro teve como inspiração um livro francês disponível no Imperial Instituto de Surdos-Mudos e que acaba confirmando a influência da língua de sinais francesa sobre a língua de sinais que se consolidava no território brasileiro.

Em 1968, a publicação de um artigo de J. Kakumusu intitulado “Urubu Sign Language”, relatou a existência de outra língua de sinais no Brasil, utilizada pelos índios Urubus-Ka’apor localizados no estado do Maranhão (ALMEIDA, 2012). Nessa comunidade indígena a percentagem relativamente alta de pessoas surdas levou a tribo a desenvolver uma língua de sinais própria utilizada por todas as pessoas da comunidade, seja ela surda ou não.

A partir da década de 1980, tem-se início discussões sobre o bilinguismo no Brasil. As pesquisas realizadas pela linguista Lucinda Ferreira Brito acerca da Língua Brasileira de Sinais, definida até então como LSCB (Língua de Sinais dos Centros Urbanos) para diferenciar da língua de sinais usada pelos índios Urubu-Ka’apor, contribuíram para a consolidação do reconhecimento legal da língua. (FIGUEIRA, 2011).

A Língua Brasileira de Sinais foi denominada Libras a partir do II Congresso Latino Americano de Bilinguismo para Surdos, realizado em 1993, em substituição à denominação LSCB, posto que LSCB fosse o termo utilizado apenas em pesquisas linguísticas enquanto Libras é o termo utilizado pela comunidade surda (ALMEIDA, 2012).

O reconhecimento legal da Língua Brasileira de Sinais como língua no Brasil é recente. A Libras foi reconhecida como língua oficial das comunidades surdas no Brasil pela Lei 10.436, de 24 de abril de 2002. Em seu Artigo 1º a Lei profere: “É reconhecida como meio legal de comunicação e expressão a Língua

Brasileira de Sinais - Libras e outros recursos de expressão a ela associados”. Em parágrafo único complementa:

Entende-se como Língua Brasileira de Sinais - Libras a forma de comunicação e expressão, em que o sistema linguístico de natureza visual-motora, com estrutura gramatical própria, constitui um sistema linguístico de transmissão de ideias e fatos, oriundos de comunidades de pessoas surdas do Brasil.” (BRASIL, 2002).

A regulamentação da Lei se deu por meio do decreto nº 5.626 de 22 de dezembro de 2005. O decreto garante o direito de inclusão das pessoas surdas na rede de instituições de ensino federal e determina a obrigatoriedade da inclusão da Libras como disciplina em cursos de Licenciatura.

### 2.1.2 Aspectos Estruturais da Libras

Segundo Ferreira Brito e Langevin (1995), as línguas de sinais utilizam de unidades de linguagem sem significado em si na construção dos sinais. Isso equivale aos fonemas das línguas orais, que são os sons das letras e sílabas produzidos pela passagem de ar pela laringe, nariz e boca. Nas línguas de sinais, a estrutura fonológica se organiza a partir de parâmetros visuais. Assim, cada sinal que pertença ao léxico (conjunto de palavras de uma determinada língua) utiliza os parâmetros considerados gramaticais e aceitos como corretos dentro dessa língua. Ferreira considera ainda que são quatro os parâmetros principais da Libras: Configuração de mão, Movimento, Ponto de Articulação e Orientação.

Configuração da(s) mão(s) (CM) são os formatos que as mãos podem apresentar durante a realização de um sinal. A Libras possui, de acordo com Ferreira-Brito (1995), quarenta e seis configurações de mão principais, ilustradas pela Figura 1 e estando agrupadas de acordo com suas semelhanças (FERREIRA-BRITO; LANGEVIN, 1995). Na Figura 2 são representados os sinais correspondentes das palavras “telefone” e “branco”, utilizando, respectivamente, as configurações de mão de ‘Y’ e ‘B’ (coluna 12 e 1 na Figura 1, respectivamente).

1	2	3	4	5	6	
7	8	9	10	11	12	
13	14	15	16	17	18	19

Figura 1 - As 46 configurações de mão da Libras.

Fonte: Ferreira<sup>1</sup> (2010, p. 220).

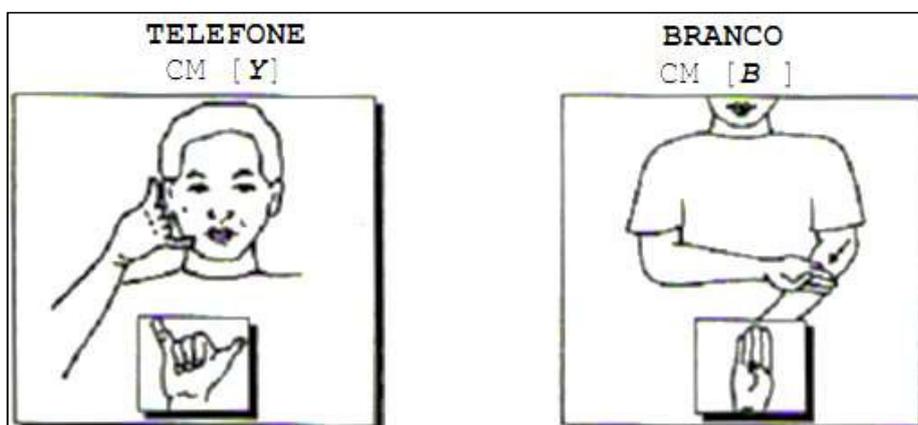
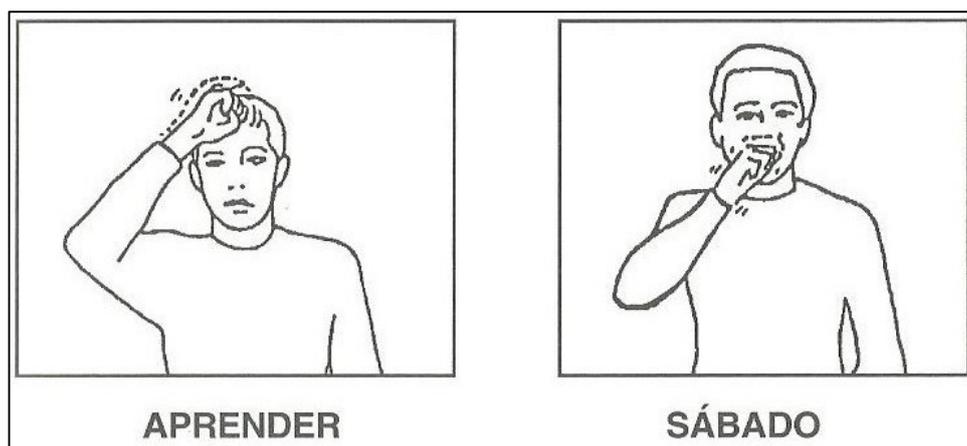


Figura 2 - Os sinais Telefone e Branco e suas configurações de mão.

Fonte: adaptado de Ferreira Brito e Langevin (1995).

<sup>1</sup> O Capítulo onde se encontra a imagem foi concebido em conjunto com Remi Langevin

O ponto de articulação é o espaço em frente ao corpo ou uma região do corpo, onde os sinais são articulados (FERREIRA, 2010). Na Figura 3 são apresentados os sinais correspondentes às palavras laranja e aprender. Ambos os sinais utilizam a mesma configuração de mão, porém, o ponto de articulação do sinal distingue os dois sinais.

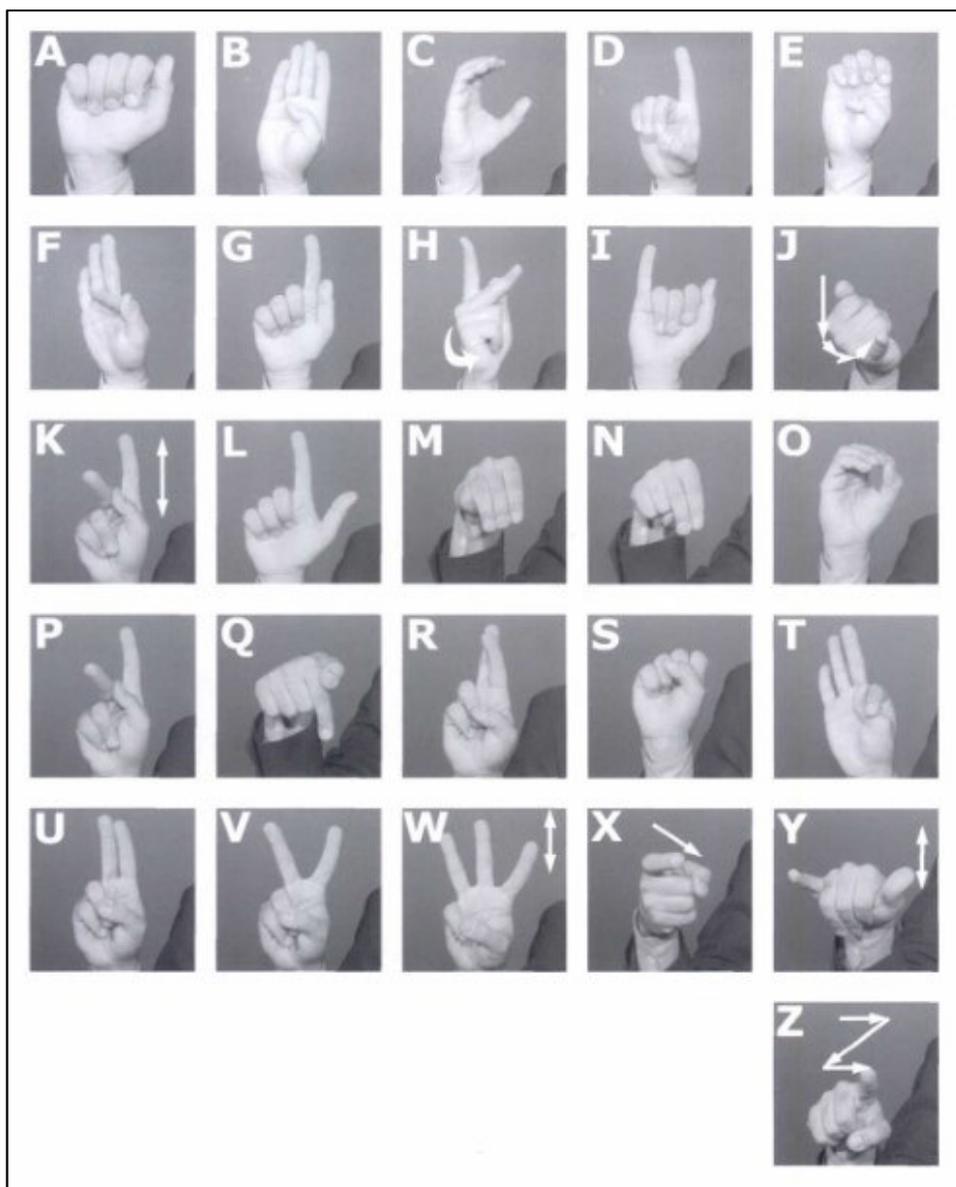


**Figura 3 - Os sinais de “aprender” e “sábado” e seus pontos de articulação**  
Fonte: Ferreira (2010, p. 39).

Movimento é um parâmetro que pode envolver um variado número de formas e direções, incluindo movimentos do pulso, movimentos retilíneos no espaço, movimentos circulares e espirais e até combinações de movimentos no mesmo sinal. Alguns sinais se diferenciam pelo movimento. A Figura 4 apresenta os sinais das palavras verde e gelado que possuem a mesma configuração e o mesmo ponto de articulação (o queixo), e diferentes tipos de movimento (FERREIRA, 2010).

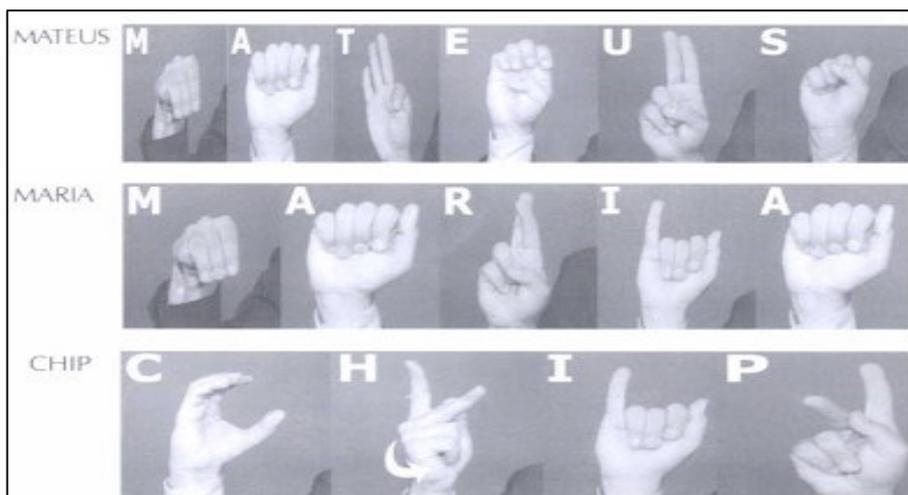
Parâmetros não manuais como as expressões faciais e movimento do corpo também desempenham papel importante na Libras. Segundo Castro e Carvalho (2009), expressões faciais e/ou corporais se fazem necessárias para que um determinado sinal tenha o significado correto. Por exemplo, o sinal de tristeza demanda uma expressão facial de tristeza.





**Figura 5 - Alfabeto Manual.**

Fonte: Castro e Carvalho (2009, p. 30).



**Figura 6 - As palavras “Mateus”, “Maria”, e “chip” em datilologia.**  
**Fonte: adaptado de Castro e Carvalho (2009, p. 31).**

#### 2.1.4 A Datilologia e o Aprendizado da Libras

Para Santana (2007, p. 233) embora seu formato seja diferente das línguas orais, o alfabeto manual se caracteriza como “uma ponte entre a linguagem oral e a língua de sinais”. Por ser o alfabeto manual é um tipo de sistema manual que representa a ortografia da linguagem oral, ele acaba se tornando um primeiro contato entre pessoas que iniciam estudos das LS.

No aprendizado da Libras por ouvintes, segundo (LEITE; MCCLEARY, 2001), alguns dos principais fatores que dificultam a aprendizagem são a modalidade da língua, a datilologia e os sinais não manuais. Parte da dificuldade na aprendizagem de línguas de sinais por ouvintes está relacionada à diferença entre línguas como o Português, que se apoiam fortemente na audição, e línguas como a Libras, que se apoiam estritamente na visão. Segundo os autores, as LS exigem um refinamento da visão para que se consiga captar todos os elementos que compõem os sinais.

Leite e McCleary (2001) afirmam ainda que a datilologia se trata de um elemento de facilidade ilusória no aprendizado da Libras, considerado como “um aspecto linguístico que não exige maior atenção nos cursos de Libras” devido a maneira relativamente rápida com que é aprendida. Apesar disso, os autores salientam que o “uso fluente da datilologia no ritmo natural do discurso espontâneo é

um dos aspectos mais difíceis de serem alcançados pelos ouvintes, exigindo uma prática muito maior do que se costuma pressupor”.

Tal recurso precisa ser reconhecido como um das dimensões da aprendizagem da Libras mais complexas para os alunos ouvintes, tanto no âmbito da produção quanto da recepção. Não basta designar uma ou duas aulas a esse tema, sugerindo (equivocadamente) que o seu ritmo natural seja minimamente próximo ao ritmo empregado por alunos iniciantes, ou que a sua função seja restrita à soletração de nomes próprios e pedidos de esclarecimentos sobre sinais da Libras. Por isso, atividades regulares, que busquem conduzir progressivamente o aluno ao uso datilológico característico do discurso espontâneo, são imprescindíveis para qualquer curso de Libras. (LEITE; MCCLEARY, 2001)

## 2.2 MÉTODOS AUTOMATIZADOS PARA RECONHECIMENTO DE LÍNGUA DE SINAIS

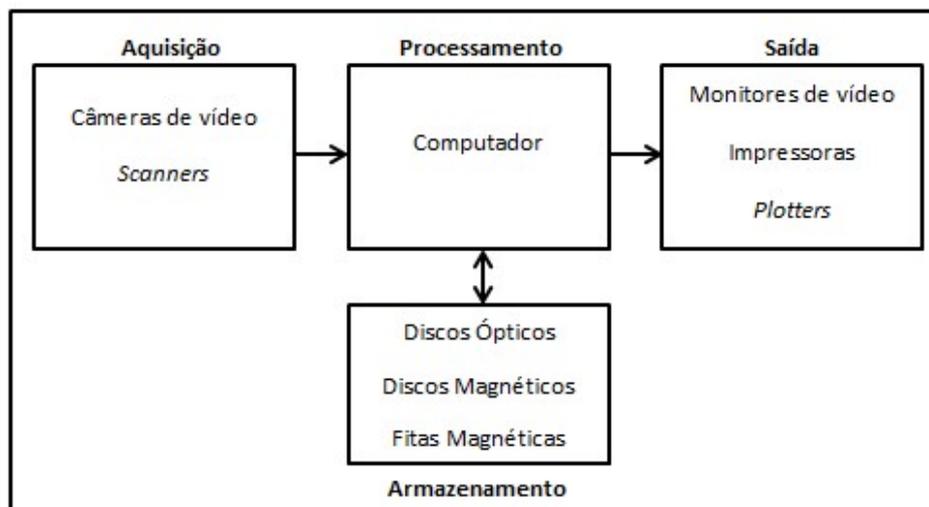
Dentre os meios mais utilizados de reconhecimento de LS podem-se citar as abordagens por visão computacional e por luvas de dados. Os métodos baseados em visão computacional simulam a visão de um observador externo a pessoa sinalizadora, enquanto os métodos baseados em luvas sensoras trabalham coletando dados dos movimentos da mão do sinalizados através de dispositivos usados pela própria pessoa que sinalizará os sinais.

### 2.2.1 Método Baseado em Visão Computacional

Os métodos baseados em visão computacional utilizam técnicas de processamento digital de imagens. A entrada para os sistemas baseados em imagens é um conjunto de imagens estáticas ou em uma sequência de vídeo necessitam que uma pessoa execute os sinais na frente de uma câmera para que um *software* possa, em seguida, interpretar e traduzir os sinais para outras línguas orais (KHAN, *et al.*, 2009).

O diagrama da Figura 7 mostra os elementos de um sistema de processamento de imagens genérico e abrange as principais operações que se pode efetuar sobre uma imagem que são: aquisição, armazenamento, processamento e

exibição. A partir das imagens capturadas da câmera, os sinais podem ser classificados por diferentes algoritmos.



**Figura 7 - Elementos de um sistema de processamento de imagens**  
 Fonte: adaptado de Filho e Neto (1999, p. 32).

O reconhecimento de gestos baseado em imagem consiste em cinco etapas: aquisição da imagem, pré-processamento, segmentação, extração de características, e classificação (FILHO; NETO, 1999).

As técnicas baseadas em imagem exibem certo número de desafios. Estes incluem: condições de iluminação, imagem de fundo, rosto e segmentação mãos, e diferentes tipos de ruído, entre outros. Mesmo que a segmentação das mãos e rosto é computacionalmente caro, os recentes avanços na computação e algoritmos tornaram possível realizar esta segmentação em tempo real (KHAN, *et al.*, 2009).

O método de reconhecimento de sinais e configurações de mão possuem os problemas e desafios inerentes dos sistemas de visão computacional, pois necessitam de características presentes na percepção visual humana, que são (FILHO; NETO, 1999): uma base de dados rica, altíssima velocidade de processamento e capacidade de trabalhar sob condições muito variadas. De acordo com Filho e Neto (1999),

o grande desafio permanece sendo o de fazer com que os sistemas de visão artificial trabalhem em diferentes condições de luminosidade, contraste, posicionamento relativo dos objetos em uma cena, sem perder a

capacidade de interpretar a cena de forma análoga à nossa capacidade de reconhecer um amigo ou parente com relativa facilidade (...)

Assim, a utilização de técnicas baseadas em visão computacional para o reconhecimento de gestos em línguas de sinais torna-se pouco prático para utilização no dia a dia, pois necessita de equipamentos de captura de imagens com certo grau de qualidade para que o resultado seja satisfatório ou é necessária utilização de várias câmeras (OZ; LEU, 2011). Tipicamente, isto requer um sistema de visão em três dimensões, utilizando de várias câmeras e demandando de métodos complexos de processamento de imagens, além da dependência das condições do ambiente como a iluminação, a cor do fundo e a roupa da pessoa (MARQUES FILHO; VIEIRA NETO, 1999).

Sistemas de detecção de cor da pele são afetados pela iluminação variável, fundo complexo, etnia do signatário (cor da pele), e oclusão do articulador do sinal. Este sistema pode parecer adequado para laboratório experimental devido à iluminação rigidamente controlada mas vai resultar em uma taxa de erro grande se utilizada em locais não adequados (KHAN, *et al.*, 2009).

## 2.2.2 Método Baseado em Luva Sensora

Métodos de reconhecimento baseados em luva sensora (ou luva de dados) constituem-se em processar os dados adquiridos de luvas equipadas com sensores (MOHANDÉS; DERICHE; LIU, 2014). Estas luvas podem fornecer informações sobre a posição, rotação, movimento, orientação da mão e flexão dos dedos. Um grande número de características pode ser extraído a partir dos dados adquiridos a partir das luvas. Estas características podem ser usadas como dados de entrada para um algoritmo de classificação para reconhecer o sinal articulado.

De acordo com Fahn e Sun (2005), a luva de dados é um dispositivo multissensorial, que gera uma quantidade significativa de dados a respeito da postura da mão. Os autores ainda afirmam que, apesar da tecnologia não ser recente, pesquisadores ainda adotam este tipo de dispositivo, devido à característica de interface natural da luva de dados com o ser humano e consideram que seja o

caminho mais adequado para melhorar a manipulação de sistemas que são aplicáveis em muitos campos específicos.

A Sayre Glove, criada pela Electronic Visualization Laboratory em 1977, foi a primeira luva sensora a ser desenvolvida. Esse trabalho pioneiro inspirou a indústria de realidade virtual que passou a desenvolver outros modelos de luvas. A empresa Nintendo utilizou a Power Glove, fabricada pela Mattel, em alguns jogos para o console SNES, porém, fracassou nas vendas por apresentar diversos problemas. Empresas mais recente como a CyberGlove e a 5DT produzem diversos modelos de luvas com número variado de sensores (STURMAN; MEDIALAB; ZELTER, 1994). A Figura 8 mostra um modelo de luva fabricada pela CyberGlove que faz uso de 18 sensores.



**Figura 8 - CyberGlove com 18 sensores.**

Fonte: Oz e Leu (2011, p. 1206).

Os sensores mais comumente utilizados para reconhecimento de gestos e configuração de mão aplicada para o desenvolvimento de dispositivos de entrada à base de luvas sensoras são os sensores de contato, sensores de aceleração, sensores resistivos de flexão e sensores de posição magnéticos.

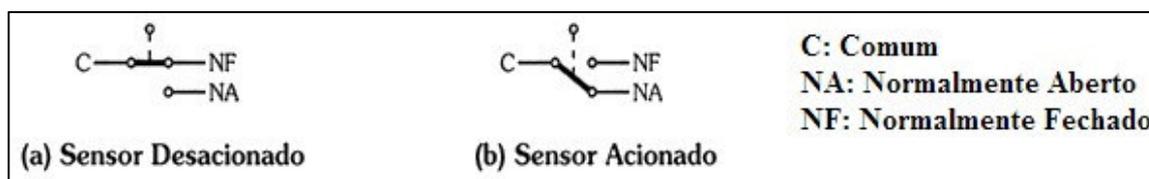
O desempenho e a eficácia destes métodos são fortemente dependentes da densidade de sensores, por exemplo, mais sensores podem ser adicionados para medir as curvas do cotovelo. O uso de mais sensores em um único projeto pode aumentar a carga sobre o processador demandando de recursos de *software* mais avançados para os intervalos de leitura dos sensores e gerenciar o consumo de energia (FAHN; SUN, 2005).

## 2.3 SENSORES

Esta seção visa introduzir alguns tipos de sensores utilizados na fabricação de luvas sensoras, apresentando seus princípios de funcionamento e aplicações. São apresentados quatro tipos: os sensores de contato, sensores de aceleração, sensores resistivos e sensores magnéticos.

### 2.3.1 Sensor de Contato Mecânico

Sensores de contato mecânico podem ser considerados simplesmente como uma chave elétrica (BONACORSO; NOLL, 1997). O sensor de contato apresenta três terminais acessíveis, sendo o comum, o normalmente aberto e o normalmente fechado, conforme a Figura 9.



**Figura 9 - Esquema elétrico dos terminais do sensor de contato mecânico.**

Fonte: adaptado de Bonacorso e Noll (1997).

Os sensores de contato são utilizados na área da robótica e automação industrial principalmente como sensores de fim de curso. Por exemplo, uma aplicação seria verificar quando uma parte da máquina ou do robô atinge uma

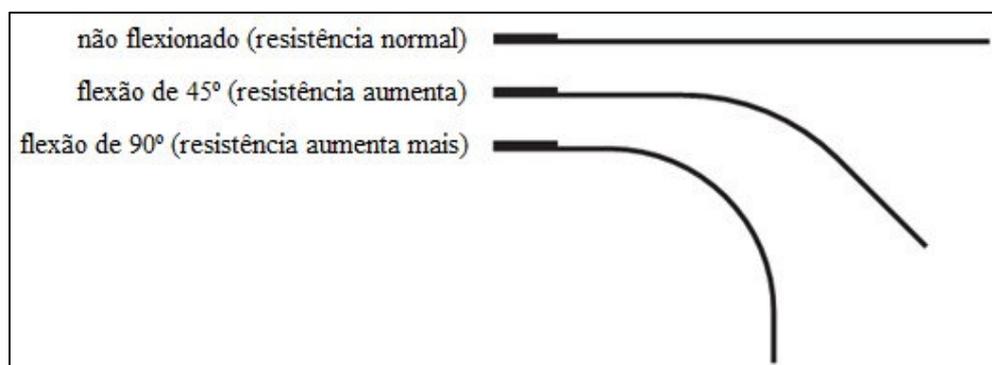
posição limite. O contato da parte com o sensor gera a informação necessária para o controlador deter o movimento dessa parte (PAZOS, 2002).

Esse tipo de sensor apresenta uma resposta binária, onde apenas dois tipos de leitura são possíveis que são contato fechado e contato aberto. Na utilização para o reconhecimento das configurações de mão esses sensores são aplicáveis nas laterais dos dedos, para indicar se estão juntos ou separados dos outros dedos quando a mão estiver aberta. Para outros parâmetros, como o ângulo dos dedos, esse sensor não apresenta resposta (TANYAWIWAT; THIEMJARUS, 2012).

### 2.3.2 Sensor Resistivo de Flexão

De acordo com Bolton (BOLTON, 2005), se uma peça de fio metálico é tracionada, não somente se torna mais longa e estirada, como também sua resistência aumenta. Quanto maior o esforço sofrido pelo fio, tanto maior o aumento da resistência.

Os sensores de flexão patenteados pela empresa Spectra Symbol são sensores que utilizam do princípio anteriormente descrito. O sensor consiste de um substrato flexível que serve de base para uma placa de material condutor. Quando o sensor é flexionado, a resistência da placa metálica varia entre 60 k a 110 k Ohms. Com base nessa variação de resistência é possível estimar qual é o ângulo em que o sensor foi flexionado (SPECTRA SYMBOL, 2014). A Figura 10 ilustra o funcionamento do sensor de flexão.



**Figura 10 - Funcionamento do Flex Sensor.**

**Fonte: Specta Symbol (2014, p. 1).**

### 2.3.3 Sensores de Aceleração

Segundo Rocha (2014), a aceleração é uma grandeza física cinemática que mede quão rapidamente a velocidade de um corpo varia ao longo do tempo e o sensor utilizado para medir essa grandeza é chamado acelerômetro. Diversos dispositivos eletrônicos fazem uso do acelerômetro como, por exemplo, em *notebooks* para proteção de discos rígidos e em *tablets* para identificar a orientação da tela.

Os acelerômetros, de acordo com Duarte (2013), são divididos em dois grupos: os mecânicos e os eletromecânicos. No primeiro grupo os acelerômetros mais comuns são os capacitivos, os piezoelétricos e os piezo-resistivos enquanto que no segundo grupo, destacam-se os sensores micro eletroeletrônicos (MEM).

A tecnologia MEM possibilita a montagem de estruturas mecânicas e eletrônicas em circuitos integrados de grande desempenho. Essas estruturas incluem capacitores internos ao módulo que contém partes móveis que por inércia resistem ao movimento. Nos acelerômetros MEM capacitivos são montadas três placas que constituem dois capacitores com uma placa central móvel. Conforme se estabelece o movimento acelerado na direção considerada, a distância entre as placas se altera e a capacitância dos dois capacitores (1 e 2) varia. Ao monitorar a variação da capacitância em função da rapidez de movimentação da placa móvel, o dispositivo pode medir a aceleração em função do tempo. Um processador eletrônico monitora essas variações de capacitância e obtém a aceleração (DUARTE, 2013).

### 2.3.4 Sensores Indutivos

Para aplicação em luva sensor, os sensores indutivos são desenvolvidos para determinar a distância e o ângulo do sensor em relação a uma fonte de campo magnético através do princípio de indução magnética (FAHN; SUN, 2005). Os sensores são constituídos de pequenas bobinas de fio de cobre e são posicionados na luva em regiões onde se deseja calcular a distância dos sensores em relação às bobinas geradoras de campo.

As bobinas geradoras são ativadas pelo sistema da luva emitindo um sinal eletromagnético e as bobinas sensores dispostas sobre os dedos são usadas para detectar esses sinais. Com base nos sinais medidos, os ângulos de flexão das articulações dos dedos podem ser calculados pelas equações formuladas nos artigos publicados por Fahn e Sun.

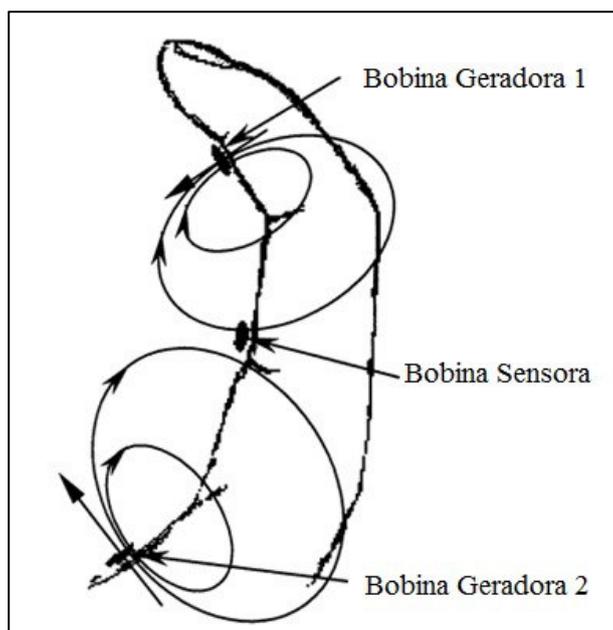


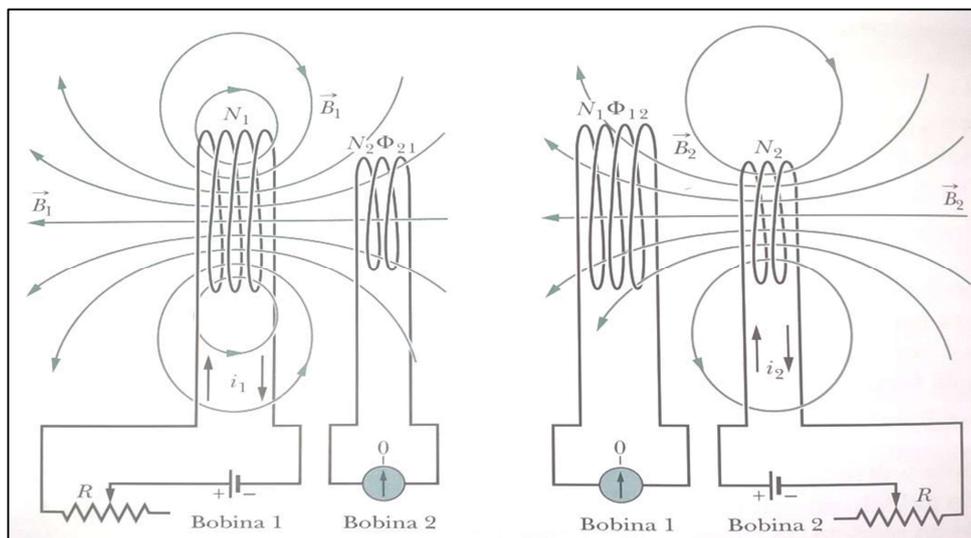
Figura 11 - Exemplo de posicionamento das bobinas.  
Fonte: Adaptado de Fahn e Sun (2005, p. 589).

Segundo Halliday *et al.* (2009), uma força eletromotriz, ou FEM, pode ser induzida em uma espira fazendo variar a quantidade de campo eletromagnético que atravessa a espira. Os valores da FEM e da corrente induzida são determinados pela taxa de variação do número de linhas de campo magnético que atravessam a espira. Assim, a FEM induzida na segunda bobina é determinada pela Equação 1, onde  $\frac{d\Phi}{dt}$  é a taxa de variação com o tempo do fluxo magnético ( $\Phi$ ) que atravessa um número  $N$  de espiras.

$$\mathcal{E} = - N \frac{d\Phi}{dt} \quad (1)$$

Quando a indução ocorre entre duas bobinas o processo de indução é denominado de indução mútua, ilustrado na Figura 12. Uma corrente  $i$  em uma das bobinas faz com que um fluxo magnético  $\Phi_{21}$  atravesse a outra, enlaçando as duas

bobinas. Se a corrente  $i$  varia com o tempo, uma força eletromotriz  $E$  dada pela Lei de Faraday aparece na segunda bobina.



**Figura 12 - Indução mútua**

Fonte: Adaptado de Halliday et al (2009, p. 272).

A indução mútua da bobina 2 em relação à bobina 1 é dada pela Equação 2, onde  $N_2$  é o número de espiras da bobina 2,  $\Phi_{12}$  é o fluxo magnético transpassando a bobina 2 devida à corrente  $i_1$  da bobina 1. Quando se faz variar a corrente  $i$  na bobina 1 (fazendo variar a resistência  $R$ ), uma força eletromotriz é induzida na bobina 2 cujo valor é dado pela Equação 3.

$$\mathcal{E}_{21} = - N_2 \frac{d\Phi_{12}}{dt} \quad (2)$$

$$\mathcal{E}_{12} = - N_1 \frac{d\Phi_{21}}{dt} \quad (3)$$

Esse princípio da indução é aplicado comumente em transformadores, empregados em circuitos de corrente alternada para aumentar ou diminuir uma tensão. Uma corrente variável em uma das bobinas do transformador produz uma força eletromotriz alternada na outra bobina (YOUNG; FREEDMAN, 2009).

Um dos problemas envolvidos no projeto de transformadores é que a potência fornecida da bobina de fonte depende da resistência da fonte da carga do

circuito conectado à outra bobina. A transferência de potência do circuito é máxima quando ambas as impedâncias são iguais.

Uma maneira de garantir essa transferência máxima de energia é adicionar um capacitor em série ou em paralelo às bobinas para que o conjunto possa estabelecer uma ressonância. De acordo com Young e Freedman (2009, p. 365) a ressonância é um fenômeno onde a impedância do sistema é mínima para determinada frequência angular. Para um circuito constituído de uma bobina e um capacitor em série ou paralelo essa frequência de ressonância é determinada pela Equação 4, onde  $f$  é a frequência de ressonância,  $L$  é a indutância da bobina e  $C$  é a capacitância do capacitor.

$$C = 1 / \omega^2 L^2 \quad (4)$$

Para o projeto dos sensores e geradores deste trabalho, uma forma de cálculo da ressonância dos circuitos é fixar um valor para a frequência e para a bobina e calcular o capacitor apropriado para que o circuito atinja a ressonância. Reformulando a Equação 4 temos que a capacitância para o capacitor do circuito é dado pela Equação 5.

$$C = 1 / \omega^2 L^2 \quad (5)$$

## 2.4 CONDICIONAMENTO DE SINAIS

Nesta seção são introduzidos os circuitos utilizados para realizar o condicionamento de sinais. Segundo Pallàs e Webster (1991) unidades de condicionamento de sinais são constituídas de circuitos eletrônicos que podem realizar operações de amplificação, filtragem, casamento de impedância, modulação e demodulação.

Condicionadores de sinal podem ser descritos também como circuitos que convertem os parâmetros elétricos de saída dos sensores em um sinal elétrico adequado para a medição. Sua utilização se faz necessária quando, por exemplo, se precisa converter um sinal de um sensor cuja saída não se dá por meio de tensão

elétrica ou para incrementar a relação sinal ruído para níveis adequados (PALLÀS; CASAS; BRAGÓS, 2008).

Geralmente os sistemas de medições são digitais enquanto os sensores tem uma grandeza de saída analógica, demandando de uma unidade conversora denominada conversor analógico digital. Esse tipo de dispositivo necessita que o sinal de entrada seja em corrente contínua ou que apresente baixa variação para que a amplitude não ultrapasse sua faixa de tensão de operação. Assim os sinais precisam ser condicionados para que possam ser convertidos. (PALLÀS-ARENY; WEBSTER, 1991)

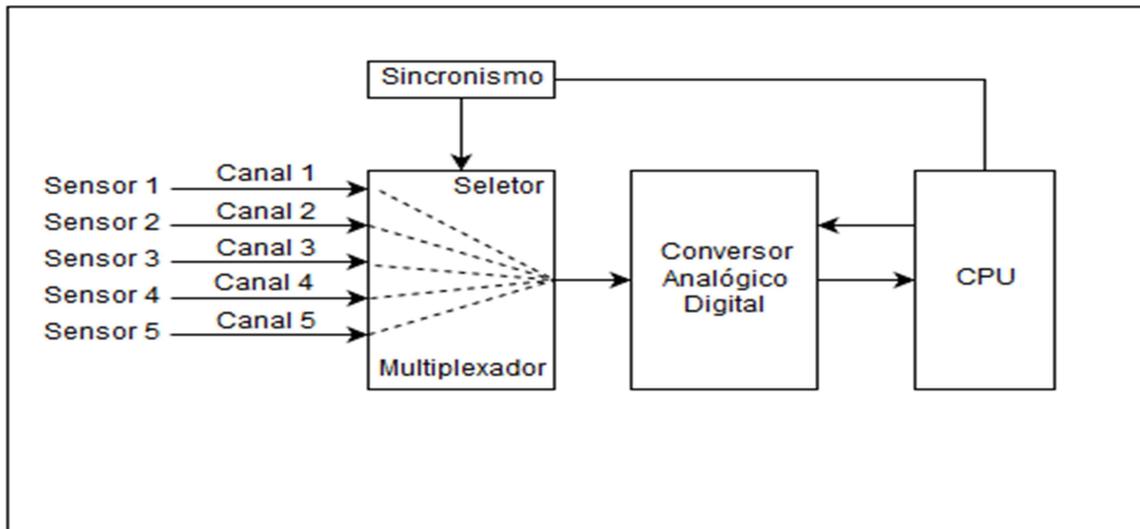
A seguir, são introduzidas algumas das técnicas de condicionamento de sinais mais comuns e que serão utilizadas no projeto do sistema.

#### 2.4.1 Filtragem

Segundo Boylestad e Nashesky (2013), filtros, no escopo eletrônico, são dispositivos que atenuam sinais em determinadas frequências. Um circuito de filtro eletrônico pode ser construído fazendo uso tanto de componentes passivos, como resistores e capacitores, como componentes ativos, que utiliza um amplificador para amplificar a tensão e isolamento do sinal.

#### 2.4.2 Multiplexagem

Multiplexagem é uma técnica que possibilita a medição de diversos sinais utilizando um único equipamento de medição. Na Figura 13 é ilustrado o diagrama de um sistema de medição utilizando multiplexagem. Na Figura 13, diversos sensores são conectados as entradas do dispositivo de multiplexação (mux). Um sinal de controle proveniente de um computador é responsável por selecionar um canal do multiplexador de cada vez (CASSIOLATO, 2012).



**Figura 13 - Multiplexagem.**

Fonte: adaptado de Cassiolato (2012, p. 1).

### 2.4.3 Retificação

Segundo (MALVINO; BATES, 2007) um sinal senoidal tem um valor médio igual nulo sobre um ciclo, em virtude da simetria do sinal em cada semiciclo, a soma de cada tensão instantânea é nula. O retificador tem então a função de contornar essa situação possibilitando uma leitura do valor médio em corrente contínua.

Em um retificador de meia onda, o diodo tem a função de cortar os semiciclo negativos resultando em um sinal de meia onda. Esta tensão de meia onda produz uma corrente unidirecional na carga, ou seja, ela circula somente em um sentido (MALVINO; BATES, 2007).

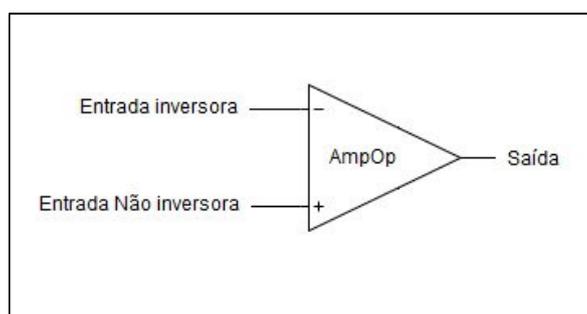
### 2.4.4 Amplificação e Isolação

De acordo com Cassiolato (2012), se em um sistema de aquisição possuir como entrada sinais de baixa intensidade, estes devem passar por uma unidade de amplificação. O sinal deve ser amplificado para que a faixa de tensão do sinal a ser medida seja semelhante à faixa de tensão suportada pelo conversor analógico digital.

Um sistema de aquisição pode apresentar sinais de corrente elevada que pode causar danos ao conversor. Para evitar essas situações, segundo Cassiolato (2012) são utilizados os *buffers* de isolamento. O *buffer* garante uma alta impedância do sinal de entrada, e baixa impedância de saída, possibilitando que o sinal a ser medido tenha sua tensão replicada na saída do *buffer*. Outra razão para a isolamento é garantir que as leituras do equipamento de aquisição serão imunes a diferenças de potencial de terra, ou tensões de modo comum (CASSIOLATO, 2012).

#### 2.4.5 Amplificador Operacional

Segundo (BOYLESTAD; NASHESKY, 2013), um Amplificador operacional ou AmpOp é um amplificador diferencial de ganho muito alto com impedância de entrada muita alta e baixa impedância de saída. Devido a suas características, os AmpOps possuem um vasto número de aplicações na eletrônica, como, por exemplo, amplificadores, osciladores, filtros e diversos outros tipos de circuitos de instrumentação. A Figura 14 mostra um simbolismo utilizado para representar os AmpOps. Entre os circuitos práticos com AmpOps, pode-se citar o amplificador não-inversor, seguidor unitário e o circuito somador.

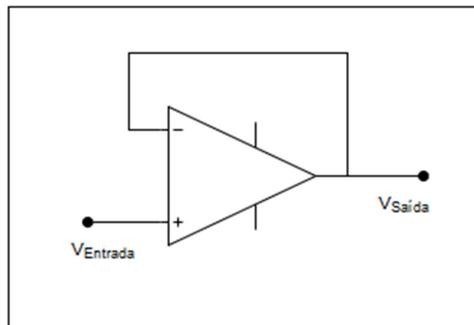


**Figura 14 - AmpOp básico.**

**Fonte: Adaptado de Boylestad, Nashesky (2013).**

Um circuito seguidor unitário, ilustrado na Figura 15, fornece um meio de isolar um sinal de entrada de uma carga ao utilizar um estágio com ganho unitário de tensão, sem inversão de fase e polaridade, e agindo como um circuito ideal com impedância de entrada muito alta e impedância de saída muito baixa. A vantagem

desse circuito é que a carga na entrada tem pouca ou nenhuma interferência na saída (BOYLESTAD; NASHESKY, 2013).



**Figura 15 - Seguidor unitário.**

**Fonte: Adaptado de Boylestad, Nashesky (2013).**

AmpOps podem melhorar o desempenho de circuitos com diodo, como por exemplo, circuitos de retificação de onda. Em primeiro lugar um AmpOp com realimentação negativa reduz o efeito da tensão de joelho, permitindo implementar retificadores, detector de pico, limitador e grampeador de sinais de pequena amplitude. E devido a ação buffer, os AmpOps podem eliminar os efeitos da fonte e da carga nos circuitos com diodo (MALVINO; BATES, 2007).

#### 2.4.5.1 Topologia de Retificador com AmpOps

A escolha da topologia do retificador levou em consideração o comparativo entre circuitos de retificação, elaborado por Dragos Ducu (2011), sendo escolhida a topologia da Figura 16. Este circuito apresenta maior precisão em relação a outras topologias, com a desvantagem do circuito possuir maior complexidade.

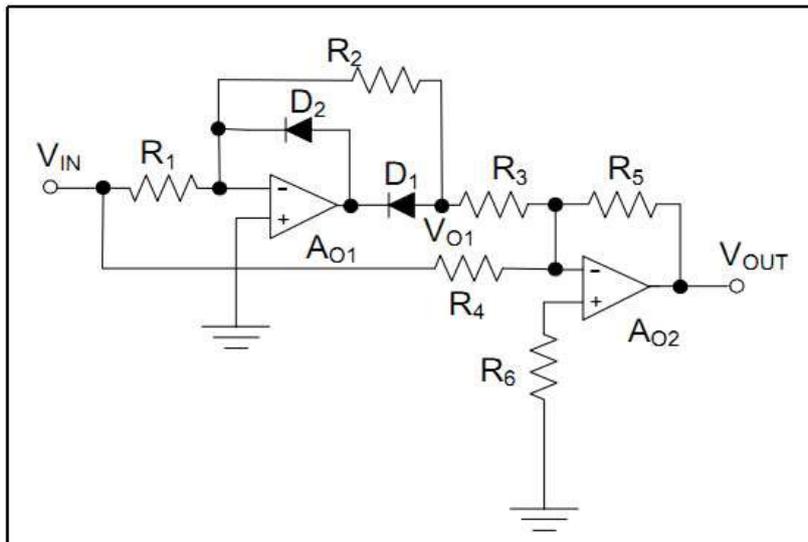


Figura 16 - Retificador onda completa de dois estágios.  
Fonte: (DUCU, 2011);

## 2.5 SISTEMAS EMBARCADOS

Sistemas embarcados são tipicamente aplicações de uma função. Várias restrições associadas a sistemas embarcados são: baixo custo, mínimo de componentes, baixo consumo, resposta em tempo real e suporte de coexistência de *software* e *hardware* (DUBEY, 2009). Uma metodologia geral usada no desenvolvimento de sistemas embarcados é mostrada no Quadro 1.

De acordo com Marwedel (2011), sistemas embarcados estão muito presentes no dia-a-dia. As áreas de aplicação são diversas, podendo-se citar a área de eletrônica automotiva em equipamento como ar-condicionado, *GPS*, sistema antifurto, freios ABS (*anti-break system*), a área da aviação em sistemas de controle de voo, sistema anticolisão, sistemas de informação do piloto, etc. na área de telecomunicação: telefonia móvel e radio amadorismo, na área de segurança em sistemas reconhecimento facial e de impressão digital.

Em questão de *hardware*, Ball (2002) afirma que os sistemas embarcados são distintos em duas categorias que são os sistemas que utilizam de microcontroladores (ou processadores *single-chip*) ou microprocessador com memória e entrada e saída externa.

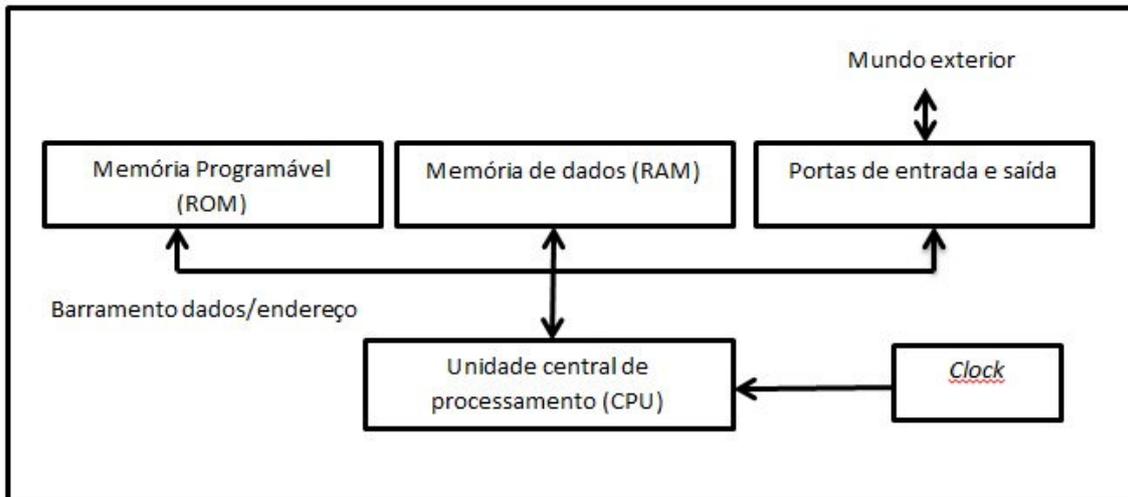
**Quadro 1 - Metodologia de desenvolvimento de um sistema embarcado.**

Fase do desenvolvimento	Detalhes da Fase
Requisitos	Requisitos funcionais e não funcionais (tamanho, peso, consumo e preço).
Especificações do usuário	Detalhes da interface de usuário combinada com as operações necessárias para satisfazer as exigências do cliente
Arquitetura	Componentes de hardware (Processador, periféricos, lógica de Programação), Componentes de software (programas principais e suas operações)
Componentes de design	Componentes modificados, novos e pré-concebidas.
Integração do sistema (hardware e software)	Esquema de verificação para descobrir erros rapidamente.

Fonte: adaptado de Dubey (2009).

### 2.5.1 Microcontrolador

O Microcontrolador ou MCU (*Micro-Controller Unit*) é um dispositivo eletrônico que tipicamente contém todas as funções necessárias para construir um sistema computacional completo, incluindo memória (DAVIES, 2008). Em geral, o microcontrolador é constituído de um Circuito Integrado (CI) que contém processador, memória, periféricos e dispositivos de interface. A Figura 17 representa um diagrama em blocos dos componentes básicos de um microcontrolador.



**Figura 17 - Componentes essenciais de um microcontrolador.**

Fonte: adaptado de Daves (2008).

A unidade central de processamento ou CPU (*Central Processing Unit*) é constituída de uma unidade lógica aritmética, registradores para a operação básica e da CPU (como o contador de programa, registrador de pilha e registradores de estados), registradores de propósito geral e decodificador de instrução. O Microcontrolador consta ainda de memórias de programa (memórias não voláteis) e memória para os dados (geralmente volátil). As portas de entrada e saída proveem comunicação digital do microcontrolador com o mundo exterior. Os barramentos de dados e de endereço interligam os componentes do microcontrolador para transferência interna de dados e instruções. O *clock* mantém o sistema todo sincronizado através de um pulso elétrico constante que pode ser gerado por um cristal interno ao MCU ou ser de fonte externa (DAVIES, 2008).

A maioria dos MCUs possuem os componentes citados mudando apenas as formas como são implementados. O que diferencia os diversos tipos de MCUs, de acordo com Martins (2005) são a quantidade de memória de programa e memória de dados, a velocidade de processamento, o número de pinos de entrada e saída, a forma de alimentação, os tipos e as quantidades de periféricos, a arquitetura e o conjunto de instruções disponibilizado (MARTINS, 2005).

### 2.5.2 Periféricos e Interfaces

Entre os periféricos mais comuns e geralmente utilizados em sistemas de luvas sensor podem-se citar os *timers*, conversores analógico/digital e interfaces de comunicação síncrona e assíncrona. As funções básicas dos *timers* referem-se à medição ou geração de eventos baseados no tempo. Os *timers* geralmente podem medir o tempo em relação a um *clock* interno do microcontrolador, embora alguns possam ser cronometrados a partir de uma fonte externa. A maioria dos microcontroladores possui pelo menos um *timer* por seu alto número de funcionalidades como provedor de uma marca de tempo regular usado para escalonar tarefas de um programa (FREESCALE SEMICONDUCTOR INC., 1998).

O conversor analógico/digital (*analogic/digital converter* - ADC) é um periférico que pode ser usado para converter o nível de tensão de sinais analógicos em um número binário que pode ser utilizado em cálculo no MCU. Muitos sinais que existem no mundo real não são diretamente compatíveis com pinos de entrada e saída de um MCU. Um conversor digital para analógico executa apenas a função oposta de um ADC, permitindo que o MCU converta um número digital em uma tensão ou corrente proporcional que pode ser usada para controlar vários dispositivos de saída em um sistema (FREESCALE SEMICONDUCTOR INC., 1998).

### 2.5.3 Desenvolvimento de *Firmware* para Microcontroladores

O desenvolvimento dos sistemas com MCUs se dá por meio de ambientes de desenvolvimento integrado (*Integrated Development Enviroment* - IDE). Segundo Davies (2008), esses ambientes contêm todas ou algumas dos seguintes itens: editor, compilador, Linker, Simulador Stand-Alone, Emulador/Depurador Embutido, Emulador *in Circuit*, Programador da memória flash. O Quadro 2 apresenta as principais características destes itens.

**Quadro 2 - As ferramentas de uma IDE.**

Editor	Usado para escrever e editar os programas. Um bom editor ajuda a organizar o código e usa cores distintas para elementos da sintaxe da linguagem.
Compilador	Produz o código executável e verifica existência de erros, preferencialmente, provendo de mensagens explicativas. Os graus de otimização podem ser alterados; tipicamente precisa-se um otimização limitada ou nenhuma durante o processo de <i>debug</i> .
Linker	Combina arquivos compilados e rotinas de bibliotecas e ajusta estes para o tipo correto de memória no MCU.
Simulador Stand-Alone	Simula a operação do MCCU em um computador <i>desktop</i> sem a necessidade de <i>hardware</i> real.
Emulador/Depurador Embutido	Possibilita a execução do código no MCU sobre o controle do Debugger em um computador <i>desktop</i> .
Emulador In Circuit	<i>Hardware</i> especializado que emula a operação do MCU sobre o controle do <i>software</i> de <i>debug</i> executando em um computador <i>desktop</i> .
Programador da Flash	Escreve o programa dentro da memória flash do MCU. Isso é feito automaticamente pelo <i>debugger</i> no desenvolvimento, porém um programador dedicado é usado na produção.

Fonte: adaptado de (DAVIES, 2008).

Quanto à linguagem de programação, existem diversas linguagens disponíveis, cada uma com suas características que as qualificam para um determinado tipo de aplicação. De uma maneira geral, as linguagens podem ser classificadas em linguagens de baixo nível e linguagens de alto nível. De acordo com Oliveira (2006) “uma linguagem é dita de baixo nível quando está na camada mais inferior do *software*”, ou ainda, é uma linguagem utilizada para interagir em nível de *hardware*. Esse nível de programação fornece a flexibilidade de operar diretamente os registradores do processador que geralmente tem seu acesso limitado pelo sistema operacional para proteger de alguma modificação inesperada.

No caso de um microcontrolador, uma linguagem de alto nível é utilizada para gerar, através de um compilador e um montador (*linker*), uma sequência de instruções e dados de programa que possam ser tratados pela CPU. Esta sequência, denominada de *firmware*, é transferida para a memória de programa do microcontrolador para que a CPU execute este programa. Entre as linguagens de alto nível pode-se dividir em linguagens estruturadas e linguagens orientadas a objeto. Nas linguagens estruturadas tem-se o Fortran, uma das primeiras linguagens de alto nível, o Pascal e o C. C é uma linguagem de programação bastante utilizada atualmente, devido a sua característica de escrever *softwares* otimizados para o processador como o Assembly, mas também possui uma série de procedimentos

prontos, como nas linguagens de alto nível. Essa flexibilidade fez com que essa linguagem de programação fosse utilizada para o desenvolvimento de uma série de sistemas operacionais modernos (OLIVEIRA, 2006).

O conceito de orientação a objetos implica na organização de um *software* em termos de objeto, incorporando suas estruturas, comportamentos e relacionamentos. Entre as linguagens orientadas a objetos, pode-se citar o C++, Visual Basic e Java.

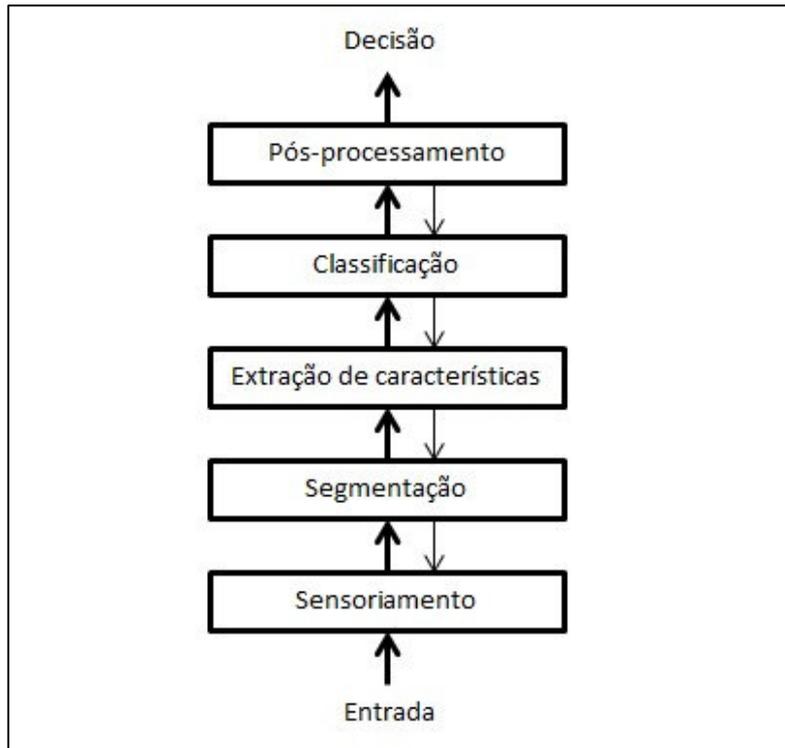
## 2.6 RECONHECIMENTO DE PADRÕES

O reconhecimento ou classificação de padrões é uma funcionalidade natural do ser humano. Com ela é possível reconhecer rostos, entender palavras ouvidas, ler textos, identificar chaves de carros no bolso pelo toque e entre outras atividades. O reconhecimento de padrões pode ser descrito como o ato de utilizar dados e realizar uma ação baseada na categoria do padrão (DUDA; HART; STOCK, 2001). Para Haykin (2001), é “o processo pelo qual um padrão ou sinal recebido é atribuído a uma classe dentre um número predeterminado de classes (categorias).”.

### 2.6.1 Sistemas de Reconhecimento de Padrões

Segundo Duda, Hart e Stock (2001, p. 8), um sistema de reconhecimento de padrões pode apresentar algumas das seguintes etapas: sensoriamento, segmentação e agrupamento, extração de características, classificação e pós-processamento. A representação de um sistema de reconhecimento de padrões é apresentada no diagrama em blocos da Figura 18.

Os dados de entrada para um sistema de reconhecimento de padrões, proveniente de algum tipo de sensor, como uma câmera ou um microfone. A dificuldade do problema pode depender das características e limitações dos sensores como resolução, sensibilidade, distorção, etc.



**Figura 18 - Diagrama de um sistema de reconhecimento de padrões.**  
**Fonte: adaptado de Duda, Hart e Stock (2001, p. 10).**

A etapa da segmentação dos dados de entrada depende do propósito do sistema. De acordo com Russel e Norvig (2004), a segmentação para sistemas de reconhecimento de fala consiste na determinação entre quando inicia e quando acaba uma palavra. No reconhecimento de padrões em imagens, a segmentação consiste no desmembramento da imagem em grupos com base na correlação entre os *pixels*.

O extrator de características indica quais as propriedades do objeto que serão úteis para realizar a classificação. O objetivo da extração de características é facilitar a tarefa de classificação indicando quais valores são similares em objetos de uma mesma categoria e quais valores são diferentes entre objetos de diferentes categorias (DUDA; HART; STOCK, 2001).

O componente que realiza a classificação dos padrões propriamente dita utiliza o vetor de características proveniente do extrator de característica para atribuir o objeto em uma categoria ou classe. Um classificador raramente é o elemento final de um sistema de reconhecimento de padrões. Após um objeto ser classificado, o pós-processamento do sistema utiliza a resposta do classificador para

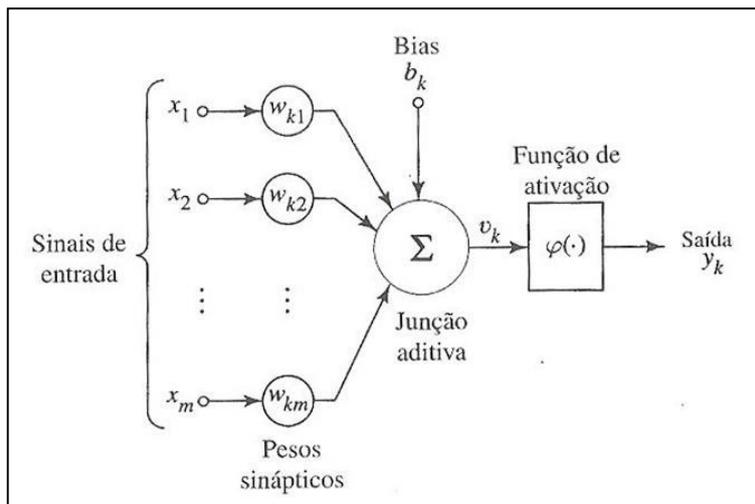
realizar uma ação relacionada com a categoria do objeto. Dentre os diferentes tipos de métodos de classificação, pode-se citar a Teoria de Decisão Bayesiana, árvores de decisão as redes neurais artificiais (DUDA; HART; STOCK, 2001).

### 2.6.2 Redes Neurais Artificiais

De acordo com Silva *et al.* (2010), redes neurais artificiais “são modelos computacionais inspirados no sistema nervoso de seres vivos”. São dotadas de capacidade de obtenção e manutenção de conhecimento e são constituídas de unidades de processamento de informação chamadas neurônios, que são interligados entre si por um grande número de interconexões (SILVA, *et al.*, 2010).

Entre as características das redes neurais mais significativas destacam-se a adaptação por experiência, capacidade de aprendizado, habilidade de generalização de conhecimento adquirido, tolerância a falhas e facilidade de implementação em *hardware* e *software*. Devido a essas características, redes neurais artificiais podem ser empregadas em diversas aplicações, tais como na aproximação de funções, no controle de processos, no reconhecimento de padrões e no agrupamento de dados (HAYKIN, 2001).

Um neurônio é uma unidade de processamento de informação com a qual é formada uma rede neural artificial. Um modelo de neurônio pode ser visualizado na Figura 19, onde é possível identificar três elementos básicos: um conjunto de sinapses ou elos de conexão, um somador e uma função de ativação (HAYKIN, 2001).



**Figura 19 - Modelo de um neurônio.**

**Fonte: Haykin (2001, p. 36).**

No conjunto de elos de conexão cada sinal de entrada ( $X$ ) conectado ao neurônio é multiplicado pelo peso sináptico ( $w_k$ ). O somador tem a função de somar os sinais de entrada ponderados pelas respectivas sinapses. A função de ativação restringe a amplitude da saída de um neurônio e, por isso também é denominada como função restritiva (HAYKIN, 2001).

Segundo Luger (2013, p. 378), além das características individuais dos neurônios, uma rede neural possui outras propriedades, tais como: arquitetura e topologia da rede, o algoritmo de aprendizagem utilizado e o esquema de codificação. Quanto à disposição dos neurônios na rede, de acordo com Facelli *et al.* (2011), os neurônios podem estar arranjados em uma ou mais camadas. Segundo esses autores:

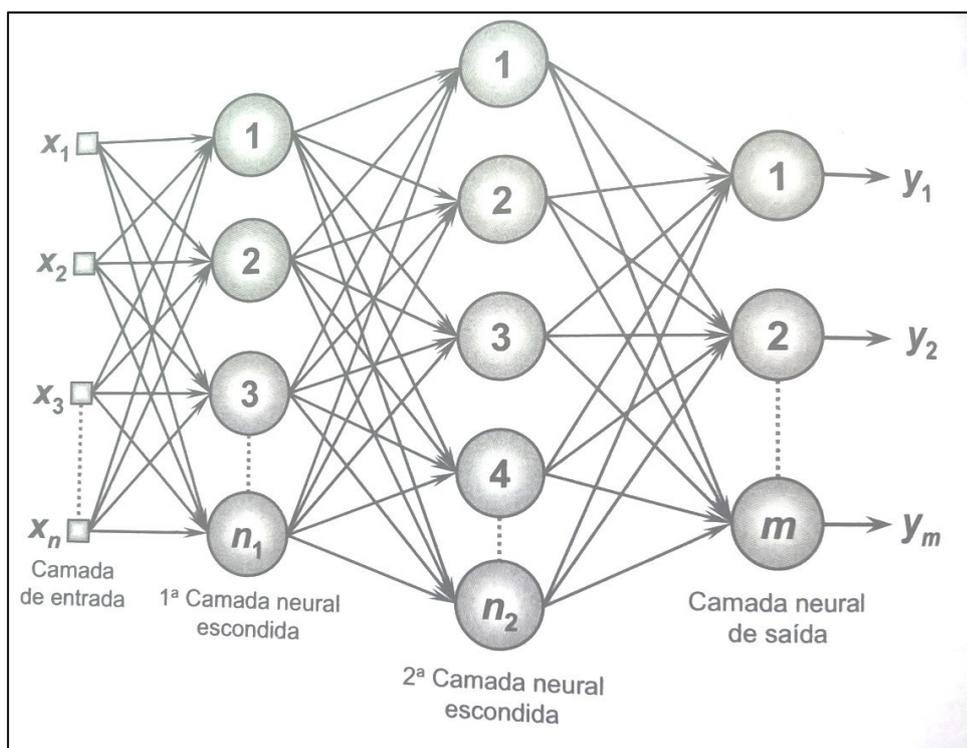
Quando duas ou mais camadas são utilizadas, um neurônio pode receber em seus terminais de entrada valores de saída de neurônios da camada anterior e/ou enviar seu valor de saída para terminais de entrada de neurônios da camada seguinte. (FACELLI, *et al.*, 2011)

### 2.6.3 Arquiteturas de Redes Neurais

Segundo Haykin (2001), entre as diversas arquiteturas de redes pode-se citar três principais, sendo estas as redes alimentadas adiante com camada única,

as redes alimentadas adiante com múltiplas camadas e as redes recorrentes. Na primeira, tem-se uma camada de nós de fonte que se projeta sobre uma camada de saída de neurônios. Nas redes de múltiplas camadas, a rede neural se distingue pela presença de uma ou mais camadas ocultas, cujos nós computacionais são denominados de neurônios ocultos. Os neurônios da camada de entrada da rede fornecem os sinais de entrada aplicados aos neurônios da segunda camada. Os sinais de saída da segunda camada são utilizados como entradas para a terceira camada e assim sucessivamente até a última camada da rede.

Segundo Silva *et al* (2010), as redes neurais alimentadas adiante de múltiplas camadas oferecem solução em diversos problemas de reconhecimento de padrões. Na Figura 20 é apresentado um modelo de rede neural de múltiplas camadas.



**Figura 20- Exemplo de rede feedforward de camada múltipla.**

**Fonte: Silva et al (2010, p. 47).**

#### 2.6.4 Aprendizagem e Algoritmo de Retro Propagação

O processo de aprendizagem ou treinamento de uma rede neural consiste em ajustar os pesos entre as conexões entre os neurônios. Dentre os algoritmos que se pode utilizar para este processo estão o algoritmo de aprendizado do perceptron (um tipo de rede de camada única), a regra delta, a retropropagação de erros e o aprendizado competitivo (SILVA, *et al.*, 2010).

Segundo Medeiros (2003, p. 10) o algoritmo de retro propagação constitui um marco no processo de aprendizagem das redes neurais, pois é capaz de convergir rapidamente para uma função. O algoritmo, também conhecido como *backpropagation*, é constituído de duas fases. Na fase primeira fase, os dados de entrada são inseridos na rede, que calcula os valores de saída. Na segunda fase, os valores são comparados aos desejados para os dados de entrada e valor do erro de cada neurônio da camada de saída é então utilizado para ajustar seus pesos de entrada. O ajuste prossegue da camada de saída até a primeira camada intermediária, propagando os erros de trás para frente (FACELLI, *et al.*, 2011).

Segundo Facelli *et al* (2011) para que o processo iterativo de treinamento da rede por retro propagação não acabe por deixar a rede ajustada para o conjunto de dados de treinamento, é necessário um critério de parada. Um critério que pode ser utilizado é a validação cruzada, que consiste em separar parte dos dados de entrada e usá-los para testar a rede durante o treinamento.

“Se as taxas de erro para os dados de treinamento e de validação forem plotadas em um gráfico, vai ser observado que no início do treinamento as duas taxas tendem a cair. Em um dado momento, a taxa de erro de validação pode começar a subir. Isso é um indício de que a rede parou de aprender e está se tornando superajustada aos dados de treinamento. Nesse ponto o treinamento da rede deve ser finalizado. Esse processo de encerrar o treinamento da rede quando a taxa de erro para o conjunto de validação começa a subir é conhecido como validação cruzada com *early stop*”. (FACELLI, *et al.*, 2011)

## 2.7 ESTADO DA ARTE

Dentre os trabalhos mais recentes sobre luva sensor, reconhecimento de configurações de mão e LS pode-se citar o artigo de Fahn e Sun (2010), cujos

autores utilizaram sensores indutivos para determinar o ângulo entre dedos da mão. A técnica de medição utilizada envolve o acoplamento magnético entre bobinas, onde o campo magnético gerado pela bobina emissora é captado pela bobina sensor, produzindo um sinal proporcional ao acoplamento entre estas. Avaliando-se esta tensão, é possível determinar o ângulo entre os dedos.

O trabalho de Silva *et al.* (2011) utiliza sensores de fibra ótica em uma luva para obter uma resposta linear dos movimentos dos dedos e, através de um programa de computador, é obtida a angulação dos dedos e uma representação tridimensional da mão. O propósito deste trabalho consiste em monitorar a eficiência de tratamentos de fisioterapia.

O trabalho de Jain *et al* (2014) utiliza sensores de flexão para determinar o movimento de dobramento dos dedos, sensores de contato na parte interna dos dedos para verificar se estes estão juntos ou separados e um acelerômetro para reconhecer as configurações de mão da Língua Americana de Sinais ou ASL (*American Sign Language*).

Técnicas de reconhecimento de gestos usando acelerômetros foram exploradas no artigo de Xu, Zhou e Li (2012). Com apenas um acelerômetro de três eixos pode-se reconhecer até sete movimentos da mão: movimentos de subir e descer a mão, para direita e esquerda, movimentos circulares e em cruz.

O artigo de Oz e Leu (2011) descreve o uso de redes neurais como método de classificação de configurações de mão da ASL. O sensoriamento da mão é realizado utilizando uma luva sensora, produzida pela CyberGlove<sup>®</sup> dispondo de 18 sensores, e um rastreador magnético de movimento Flock of Birds<sup>®</sup>.

### 3 MATERIAIS E MÉTODOS

Neste Capítulo será apresentada a metodologia para o desenvolvimento do sistema e os componentes e tecnologias que serão utilizados. Inicialmente, será descrito os materiais que foram definidos para a construção do sistema e em seguida é descrito a metodologia que será aplicada no processo de desenvolvimento do sistema.

#### 3.1 MATERIAIS

Nesta Seção será descrito os materiais que serão utilizados para o desenvolvimento do sistema. Definiu-se as luvas que serão utilizadas para fixar os sensores, os fios de cobre para a confecção dos sensores, a placa de desenvolvimento de sistemas microcontrolados, os dispositivos de interface com o usuário, além da ferramenta de desenvolvimento do código do microcontrolador e de simulação de redes neurais.

##### 3.1.1 Luva de Fixação dos Sensores

Para que o usuário do sistema tenha uma maior desenvoltura para realizar os sinais é necessário que a luva seja confeccionada de um material confortável e que garanta a flexibilidade dos dedos. Foi utilizada a luva “MaxiFlex<sup>®</sup> Ultimate<sup>®</sup>”, mostrada na Figura 21, produzida pela ATG<sup>®</sup>, uma empresa especializada em luvas de proteção individual. A luva atendeu as especificações do sistema, possuindo as características de ser flexível e por possibilitar uma boa fixação dos sensores.



**Figura 21 - Luva MaxiFlex®.**  
**Fonte - ATG Ltd (2014).**

### 3.1.2 Material para Confeção dos Sensores e Geradores

Para a fabricação dos sensores serão utilizados fios de cobre esmaltados de 2,049 mm de diâmetro. O esmalte tem a função de isolar o fio para que as diversas espiras que constituem as bobinas não entrem em contato entre si. Os capacitores utilizados têm o encapsulamento de dispositivos de montagem superficial ou SMD (*Surface Mounted Devices*).

### 3.1.3 Display

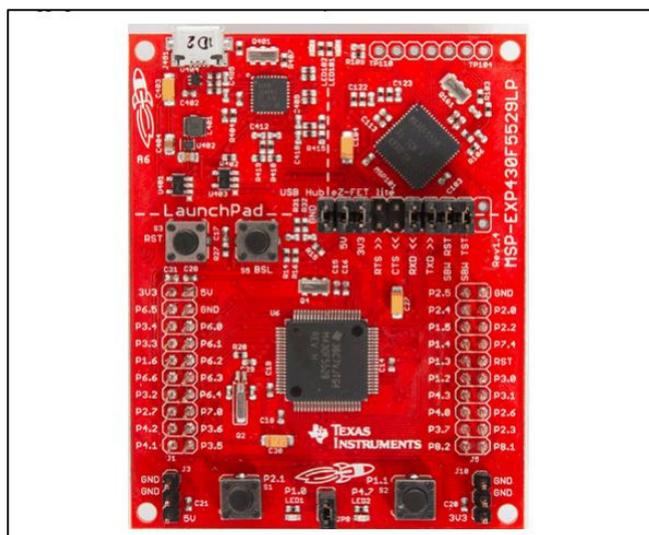
O *display* utilizado será um módulo *display* cristal líquido ou LCD (*Liquid Crystal Display*) de dezesseis caracteres por duas linhas (16x2) incluso na placa comercializado pela DFRobot®, ilustrada na Figura 22. A placa contém o módulo LCD, propriamente dito, e algumas chaves do tipo *pushbutton* que podem ser utilizados como teclado do sistema.



**Figura 22 - Módulo LCD.**  
**Fonte: DFRobot (2016).**

### 3.1.4 Placa de Desenvolvimento de Sistemas Microcontrolados

Foi utilizado a placa de desenvolvimento de sistema microcontrolados da família MSP430 desenvolvido pela Texas Instruments®, mais especificamente o “MSP430F5529 USB LaunchPad Evaluation Kit®”, ilustrada na Figura 23. A placa se mostra adequado para o sistema, pois possui os recursos necessários e capacidade de processamento adequada para a aplicação deste trabalho.



**Figura 23 - Placa de desenvolvimento MSP430F5529.**  
**Fonte: Texas Instruments Inc. (2015).**

Dentre das características da placa de desenvolvimento do microcontrolador MSP430F5529, pode-se destacar aquelas principais para a

escolha para a utilização no sistema como sendo (TEXAS INSTRUMENTS INC., 2015):

- Arquitetura RISC com CPU de 16 bits;
- Clock do Sistema de até 25 MHz;
- Tensão de operação de 1,8 V a 3,6 V;
- 128KB de memória *flash* e 8KB de memória RAM;
- Cinco temporizadores;
- Conversor analógico digital de 12 bits;
- Habilidade de emular e desenvolver aplicações USB utilizando um único cabo USB;
- Alimentação fornecida pelo conector USB. A tensão de 5 V da USB é regulada para 3.3 V, através de um regulador;
- Conector barra de pinos macho e fêmea com 40 pinos para conexão das portas com outras placas e dispositivos;

### 3.1.5 Ferramenta de Desenvolvimento de Código para o Microcontrolador

O ambiente de desenvolvimento integrado que foi utilizado para a elaboração do código do microcontrolador do sistema é o ambiente disponibilizado pela Texas Instruments® denominado “Code Composer Studio®” em sua versão 6.0.1. A empresa disponibiliza uma versão gratuita desse ambiente com limitação de tamanho de código de 16kB quando utilizado o compilador da Texas Instruments e sem limitação quando utilizado o compilador MSP430GCC. A linguagem C foi utilizada para o desenvolvimento do firmware para este microcontrolador .

### 3.1.6 Ferramenta de Desenvolvimento de Redes Neurais

O projeto e simulação da rede neural foram desenvolvidos no software MATLAB® (versão 2010), utilizando a *Neural Network Toolbox*®. Essa ferramenta inclusa no MATLAB fornece diversas funções e algoritmos para criação, treinamento, visualização e simulação de redes neurais.

### 3.1.7 Ferramenta para Desenho Esquemático e Layout de Circuito Impresso

A Placa de Circuito Impresso ou PCI onde foram implementados os circuitos de condicionamento de sinais foi desenvolvida utilizando o *software* EAGLE (*Easily Applicable Graphical Layout Editor*), desenvolvido pela empresa CadSoft®. A versão utilizada é a *Ligth Edition*, edição gratuita com limitação de tamanho de placa de 100x80mm e número de camada de sinal (duas camadas). As limitações desta versão não interferiram no desenvolvimento do *layout* da PCI.

### 3.1.8 Bibliotecas e Exemplos de Códigos

A maioria das funções e rotinas implementadas no microcontrolador provém das bibliotecas inclusas no Code Composer Studio®, como a biblioteca da família MSP430 (*mmsp430.h*), que inclui as macros de acesso aos registradores dos periféricos. Outra biblioteca do CCS utilizada foi a biblioteca de operações matemáticas otimizadas para os microcontroladores da família MSP430.

Para a utilização do display se fez uso da biblioteca de manipulação de módulo LCD com controlador HD44780 disponibilizada por Fambrini (2011).

Para o desenvolvimento do roteiro do MATLAB foi utilizado como exemplo o roteiro de aplicação da ferramenta de redes neurais para o reconhecimento de caracteres, disponível no site da MathWorks.

## 3.2 METODOLOGIA

Tendo em vista a metodologia de desenvolvimento de sistemas embarcados definida por Dubey (2009) na Seção 2.4, a primeira etapa para a realização do sistema é a definição dos requisitos. Como requisitos funcionais pode-se estipular que o sistema deve possuir um conjunto de sensores dispostos em uma luva para que seja possível determinar a posição dos dedos da mão do usuário do sistema (configuração de mão).

Um segundo requisito é que a informação sobre a configuração da mão seja decodificada como sendo uma letra do alfabeto. Um último requisito funcional é que o sistema retorne ao usuário informações sobre a utilização do sistema, como, por exemplo, informando o sucesso no reconhecimento do sinal ou informando a pontuação de acerto de uma palavra datilografada em Libras.

Um requisito não funcional adequado ao sistema é que o conjunto de sensores da luva não interfira de forma significativa na gesticulação dos sinais pelo usuário. Os componentes que farão parte do *hardware* do sistema foram mencionados na seção anterior (3.1). Os componentes, como os circuitos de condicionamento de sinais e outros circuitos de controle serão determinados a partir das exigências do sistema de sensoriamento.

### 3.2.1 Desenvolvimento do Sensoriamento da Luva

O sistema de sensoriamento da luva foi desenvolvido a partir dos artigos publicados por Fahn e Sun (2005, 2010). Nesses trabalhos realizou-se um sistema baseado em luva sensora utilizando-se de sensores magnéticos de fabricação própria. O sensoriamento da luva foi construído de maneira a utilizar um número reduzido de sensores sem perder o número movimentos detectáveis que outras luvas com mais sensores possibilitam.

Segundo Fahn e Sun (2005), a grande vantagem na utilização dos sensores magnéticos está no fato que não existe necessidade de contato do sensor com as articulações dos dedos, como acontece com sensores resistivos de flexão. Isto possibilita maior qualidade de medição do movimento e maior durabilidade da vida útil dos sensores.

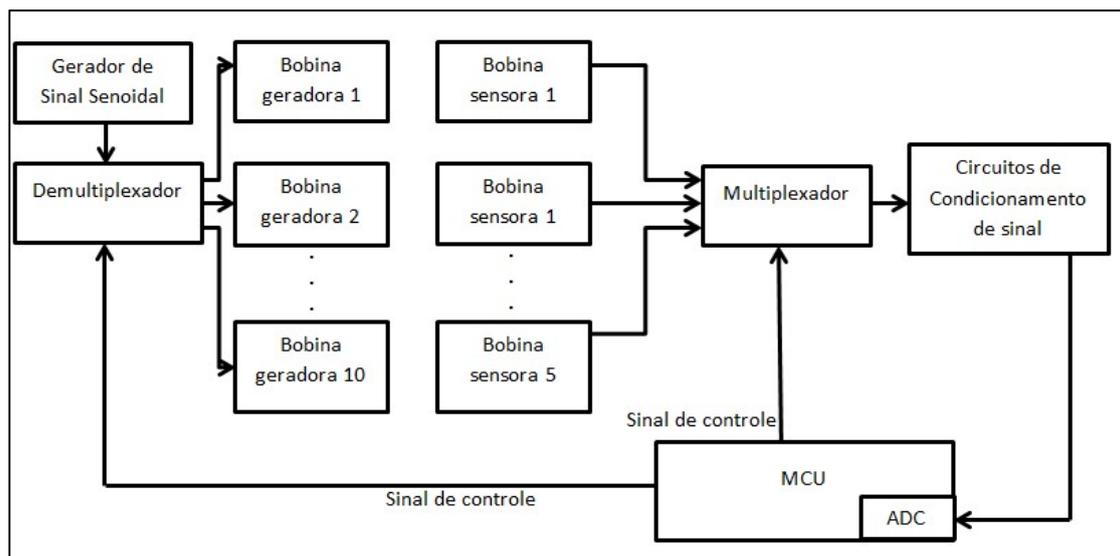
A estrutura da luva mencionada pelos autores consiste de sensores indutivos que serão excitados por bobinas geradoras de campo magnético. Os sensores são capazes de gerar um sinal elétrico induzido pelas bobinas geradoras proporcionalmente à distância e ângulo entre estes. O sinal gerado pelos sensores é tratado por um circuito de condicionamento de sinais para ser separado de possíveis ruídos.

Para este trabalho, será realizada a replicação dos sensores e geradores desenvolvidos por Fahn e Sun (2005) e realizados testes para confirmar o princípio de funcionamento do sensor. A modelagem do posicionamento dos sensores terá em vista as dimensões de uma luva de tamanho único, sendo que, a utilização do sistema por diferentes usuários com diferentes tamanhos de mão pode não levar a um mesmo resultado.

### 3.2.2 Desenvolvimento de Circuitos de Condicionamento de Sinais Analógicos

Tendo em vista que diversas bobinas geradoras serão utilizadas na luva, se faz necessário um mecanismo que previna a interferência entre as bobinas. Uma técnica para evitar a interferência mútua entre as bobinas geradoras é a utilização de diferentes frequências do sinal senoidal para cada bobina. Esse método exige que mais hardware seja adicionado ao sistema uma vez que se faz necessário diferentes fontes de sinal senoidal e filtros de passa-banda para a leitura dos sensores.

A técnica que melhor se adequa ao sistema é o método de divisão de tempo, que consiste em acionar apenas uma bobina geradora por vez. Para cada bobina geradora é atribuído um intervalo de tempo, estabelecido pelo microcontrolador, para que o gerador de sinal senoidal possa ativá-la e o MCU possa realizar a leitura do sensor correspondente. A vantagem desse método é que todo o controle do acionamento das bobinas geradoras é feito através do microcontrolador. O diagrama em blocos do controle dos sensores e geradores da luva sensora é apresentado na Figura 24.



**Figura 24 - Diagrama em blocos dos circuitos de controle da luva sensora.**

Fonte: adaptado de Fahn e Sun (2005, p. 592).

O sinal de saída do multiplexador deve passar por um processo de condicionamento. Para o sistema em questão esse processo se dará por uma série de circuitos analógicos de condicionamento utilizando amplificadores operacionais ou AmpOps. Primeiramente, o sinal de resposta dos sensores será atenuado ou amplificado para se ajustar em amplitude com os valores de tensão de alimentação dos AmpOps.

Em seguida, o sinal senoidal da saída do amplificador passará por um circuito retificação, uma vez que a informação que possibilita estimar a posição das bobinas é o módulo da FEM induzida nas bobinas dos sensores. O sinal retificado passará por um filtro para eliminar o *ripple* do sinal de corrente contínua ou CC. A saída do retificador passa novamente por ajustes de amplitude e é conectada ao pino correspondente à entrada do ADC do microcontrolador.

### 3.2.3 Desenvolvimento do Código do Sistema Microcontrolado

O código para o controle da luva terá sua estrutura do tipo laço infinito, onde as funções que desempenham os controles são executadas sequencialmente. O laço se inicia com o envio de um sinal de controle, através das portas de saída do microcontrolador, para que uma bobina geradora seja ativada. Em seguida é

realizada a leitura da porta de entrada do ADC conectada aos circuitos de condicionamento de sinais.

O microcontrolador efetua a conversão do sinal analógico e armazena a palavra digital resultante em um vetor de dados. Esse processo de seleção de gerador e leitura do ADC é realizado até que todas as bobinas geradoras sejam ativadas. Ao final desta etapa o vetor apresentará os dados provenientes dos sensores da luva.

Os dados adquiridos serão processados por um algoritmo de classificação. Os dados de resposta do algoritmo de classificação são interpretados como sendo uma letra do alfabeto. O *display* é atualizado para informar ao usuário que o sinal foi reconhecido apresentando o resultado em forma de texto.

### 3.2.4 Método para Reconhecimento de Padrões

O sistema de reconhecimento de padrão, que irá realizar a classificação dos conjuntos de dados dos adquiridos dos sensores entre as diferentes letras do alfabeto. Os dados de entrada do sistema são as posições dos dedos da mão do usuário que utiliza a luva. Um estágio de pré-processamento dos dados dos sensores pode ser realizado digitalmente no código do microcontrolador para adequar os dados obtidos, se for necessário.

O processo de segmentação dos dados consiste em determinar quando uma sequência de dados corresponde a um novo sinal. Inicialmente, este processo pode ser feito estipulando um tempo fixo para o microcontrolador realizar a leitura dos sensores, ou ainda realizar diversas classificações em um período de tempo e selecionar a classe de maior ocorrência.

Para o processo de classificação foi decidido utilizar o método de redes neurais artificiais. De acordo com Silva, *et al* (2010), as redes neurais possuem como característica a facilidade de prototipagem em *hardware* ou *software*, uma vez que, após treinadas, necessitam de operações matemáticas básicas para determinar um resultado. Também é interessante a capacidade de generalização do conhecimento da rede, possibilitando que a rede classifique as configurações de mão mesmo que

haja mudanças na maneira como o alfabeto da Libras é sinalizado (SILVA, *et al.*, 2010).

No trabalho de Oz e Leu (2011), utilizou-se de redes neurais tanto para a classificação de diversos sinais da ASL, quanto para determinar o tempo de duração de execução dos sinais. Essa aplicação das redes neurais em diferentes aspectos relacionado a dispositivos de luva sensora serviu como incentivo ao uso de desse tipo de classificador para este trabalho.

A arquitetura da rede será definida com base no conjunto de sensores da luva e das configurações de mão que esse conjunto é capaz de identificar. Para definir o número de neurônios ocultos da rede, foram definidas e testadas algumas topologias candidatas para avaliar o desempenho do reconhecimento de um conjunto de padrões de entrada.

A saída do sistema de reconhecimento de padrões fornecerá a categoria (ou classe) em que os dados recebidos se enquadram. O pós-processamento do reconhecimento de padrões será associar a classe selecionada à letra do alfabeto. A letra reconhecida é apresentada ao usuário através de um módulo LCD.

### 3.2.5 Validação e Teste do Sistema

O teste do sistema se dará, em um primeiro momento, testando individualmente os componentes do sistema. Os sensores serão testados para determinar qual a máxima distância dos geradores é capaz de ser identificada. O circuito de condicionamento será testado para verificar o nível de corrente total e o sinal de saída dos sinais dos sensores foi bem condicionado.

O reconhecimento de padrões será comparado através da simulação e do algoritmo implementado no microcontrolador para verificar se ocorreu qualquer erro de lógica ou erros devido às diferenças da arquitetura. O sistema também passa por um teste onde será sinalizado as letras do alfabeto da Libras e verificado a existência de falsos positivos no sistema de classificação ou problemas na modelagem do sensoriamento.

## 4 DESENVOLVIMENTO

Neste Capítulo são apresentadas as etapas do desenvolvimento do sistema, iniciando com uma visão geral do sistema e suas funcionalidades. Em seguida é descrita a modelagem dos componentes que constituem o sistema: a luva sensora, o circuito de condicionamento de sinais e o software implementado no microcontrolador para a aquisição dos dados. Por fim é apresentado a modelagem do sistema de reconhecimento dos padrões de configuração de mão que serão classificados de acordo com as letras do alfabeto da Libras.

### 4.1 VISÃO GERAL DO SISTEMA

A principal funcionalidade do sistema é reconhecer as configurações de mão utilizadas para sinalizar o alfabeto manual de Libras. Na Figura 25 é apresentado o diagrama de blocos do sistema.

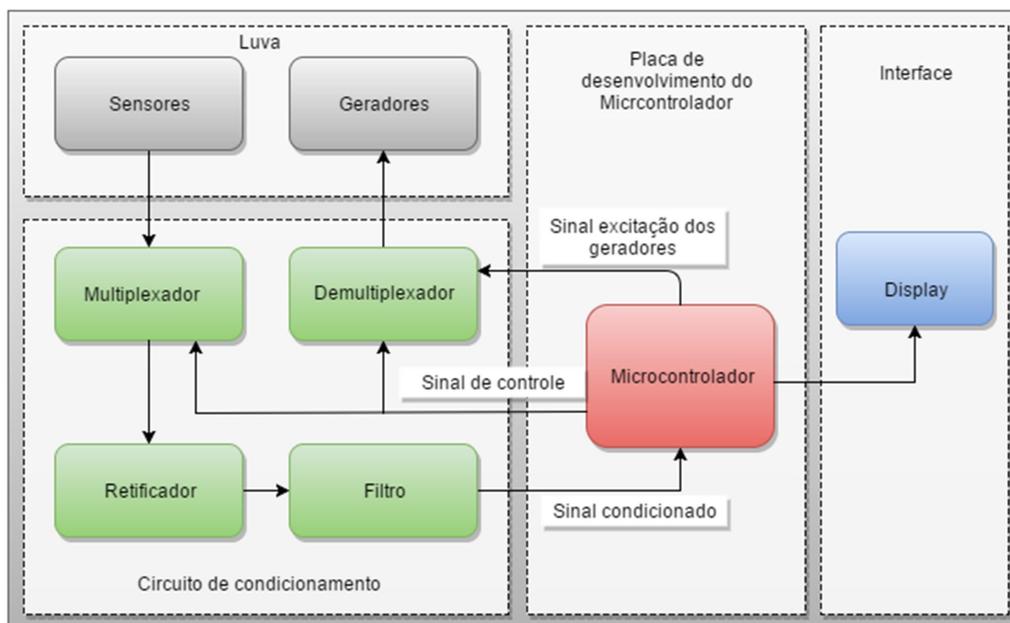


Figura 25 - Diagrama de blocos do sistema.

O sistema é composto de uma Luva munida de um conjunto de circuitos de sensoriamento (sensores e geradores), um circuito eletrônico de

condicionamento dos sinais dos sensores, um microcontrolador que irá processar os dados através algoritmo de classificação, e um display que irá apresentar ao usuário qual configuração estará sendo sinalizada.

O sensoriamento da luva se dá por meio de sensores indutivos, ou acoplamento magnético. Os sensores respondem a um campo magnético gerado por uma unidade geradora. Ambos os sensores e geradores são em suma um circuito Indutor capacitor em ressonância com a frequência de onda utilizada para alimentar os geradores.

A partir do reconhecimento das configurações de mão o sistema pode adquirir diversas funções, como, por exemplo, a tradução de palavras datilografadas em Libras ou ainda o desenvolvimento de um jogo de datilologia em Libras, com finalidade de apresentar ouvintes a Libras.

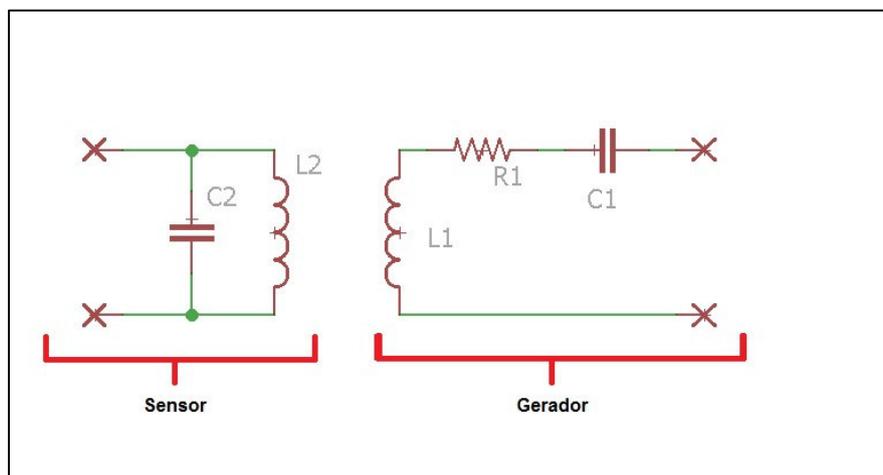
## 4.2 SENSORIAMENTO DA LUVA

Os sensores utilizados consistem em um circuito ressonante para uma frequência arbitrariamente definida. Os sensores produzem um sinal oscilatório em resposta a um sinal magnético produzido por um circuito gerador. A modelagem do sensoriamento consistiu em definir os componentes dos sensores e geradores, bem como o posicionamento destes sobre a superfície da luva. O sensoriamento da luva se dará apenas para determinar as configurações de mão, não utilizando de nenhum sensor para a detecção de movimento.

### 4.2.1 Modelagem do Sensoriamento

A topologia dos circuitos dos sensores e geradores foi definida como mostra a Figura 26. Os circuitos constituem uma espécie de transformador. Quando um sinal alternado é aplicado no circuito gerador de campo magnético, este induz uma tensão na bobina do circuito do sensor. Como o meio onde o campo magnético é gerado é apenas ar, o acoplamento magnético se torna significativamente fraco. Utilizou-se do princípio da ressonância em circuitos do tipo indutor-capacitor para

que, em determinada frequência, a corrente do gerador seja máxima. Para o projeto dos circuitos, optou-se em fixar a frequência de ressonância do sinal de excitação em 100 kHz e definir os demais componentes a partir desta.



**Figura 26 - Topologia dos circuitos de sensoriamento.**

Uma etapa essencial na modelagem do sensoriamento é a definição da quantidade de sensores presentes na luva e suas respectivas posições. Esse fator determina a complexidade do circuito eletrônico de interface da luva com o microcontrolador e também a capacidade do sistema em determinar as configurações de mão.

Quanto mais sensores a luva dispuser, mais informações sobre a posição dos dedos poderá ser obtida, porém seria necessário um tempo relativamente grande para ser efetuada a varredura destes sensores. Outro problema é que a mobilidade dos dedos pode ser prejudicada, interferindo no desempenho do usuário ao sinalizar a letra do alfabeto e, assim, gerar dados diferentes dos esperados.

Por outro lado, se o número de sensores for muito pequeno não é possível determinar as pequenas variações da posição dos dedos. Isto, conseqüentemente, dificultaria a distinção do conjunto de dados de entrada para letras que apresentam configurações de mão semelhantes. O conjunto de sensoriamento definido pode ser visualizado na Figura 27 e nos Quadros 3 e 4 são apresentadas as nomenclaturas adotadas e as posições dos geradores e sensores da luva, respectivamente.

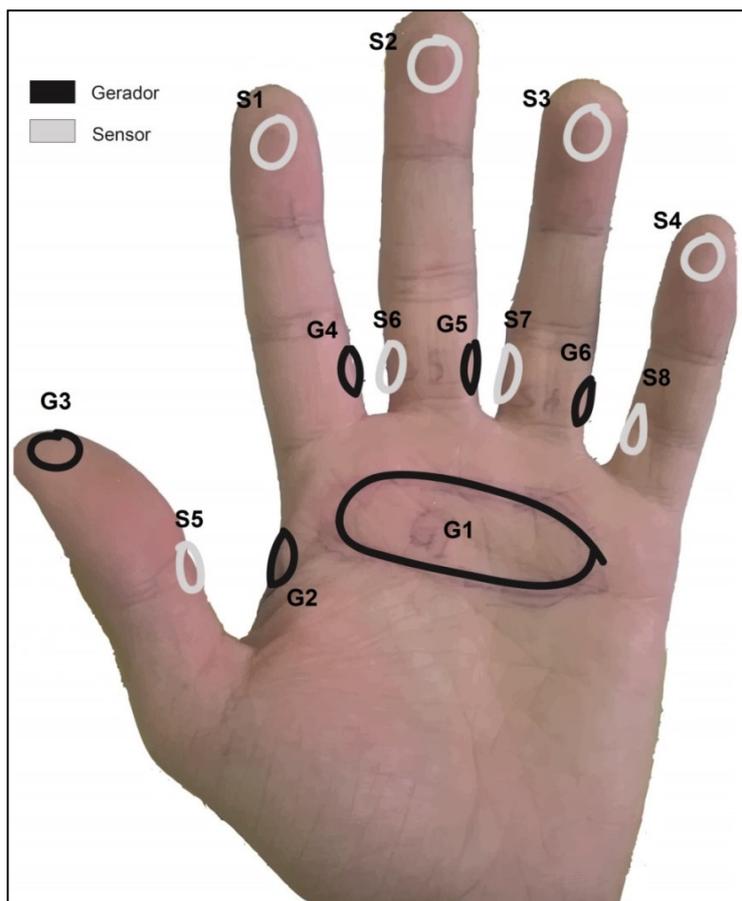


Figura 27 - Conjunto modelado de sensores e geradores da luva.

Quadro 3 - Descrição da localização dos geradores

Gerador	Localização na luva
G1	Palma
G2	Lateral da palma
G3	Extremo do polegar
G4	Lateral do indicador, entre o dedo médio
G5	Lateral dedo médio, entre o anelar
G6	Lateral dedo anelar, entre dedo mínimo

Quadro 4 - Descrição da localização dos sensores

Sensor	Localização na luva
S1	Extremidade indicador
S2	Extremidade médio
S3	Extremidade anelar
S4	Extremidade mínimo
S5	Lateral polegar
S6	Lateral médio, entre o indicador
S7	Lateral anelar, entre o médio
S8	Lateral mínimo, entre anelar

Para determinar o conjunto de sensores foi observado as diferentes configurações de mão do alfabeto da Libras e dessa forma, pode-se determinar alguns pontos de contato e proximidade entre os dedos em comum à maioria das configurações. Também foram observados pontos interessantes para a distinção de configurações semelhantes.

Por exemplo, as configurações das letras U e V distinguem-se entre si apenas pelo ângulo entre os dedos indicador e médio. Para este exemplo, a solução é posicionar um sensor em uma lateral de um dedo e um gerador na lateral do outro. Esse movimento angular dos dedos também foi observado para outras configurações, utilizando outros dedos, como as configurações da letra W e M.

Outro ponto de interesse é a palma da mão, onde existe o contato e/ou a proximidade com as pontas dos dedos em configurações como a das letras 'A', 'E' e 'S', por exemplo. Para esse ponto foi definido que existirá um gerador localizado na palma com formato que abrangerá todo o comprimento da palma.

A detecção do movimento do polegar teve que ser simplificada tendo em vista que este dedo possui uma mobilidade diferente dos outros dedos. Para este caso, o sensoriamento precisou ser tratado em quatro situações. A primeira é a proximidade e afastamento do polegar em relação à lateral da palma da mão, que servirá para a distinção entre as letras L e G, por exemplo. Um segundo caso é o posicionamento do polegar sobre a palma da mão, observado na configuração da letra B, U, V, R e W.

A terceira situação é o polegar em forma de pinça, em relação a outros dedos, como para as letras O, D e C. A última situação de posicionamento do polegar é em relação às letras P, K, H. Para esse caso, a extremidade do polegar tem proximidade com a base do dedo médio.

Para auxiliar a definição e validação do conjunto de sensoriamento, foi efetuado um mapeamento das respostas dos sensores para as configurações de mão do alfabeto. Consideraram-se três níveis de resposta dos sensores: sensor muito longe do gerador, representado pela letra L; sensor muito próximo do gerador, representado pela letra H; e sensor em uma distância intermediária do gerador, representado pela letra M.

Com essas definições, foram verificadas as relações das respostas ideais dos sensores para cada configuração de mão do alfabeto da Libras. No Quadro 5 é

exibido, como exemplo, o mapeamento para a letra A. No exemplo, podem-se obter as posições de cada sensor em relação aos geradores.

**Quadro 5 - Mapeamento dos sensores para configuração da letra A.**

LETRA: A		SENSORES							
GERADORES	Nome	S1	S2	S3	S4	S5	S6	S7	S8
	G1	H	H	H	H	L			
	G2					H			
	G3	L	L	L	L		L		
	G4						H		
	G5							H	
	G6								H

#### 4.2.2 Dimensionamento das Bobinas

As dimensões das bobinas dos circuitos de sensoriamento foram definidas arbitrariamente, visando um diâmetro que não excedesse a largura dos dedos da luva para evitar a flexão excessiva da bobina quando a luva fosse vestida, e uma espessura que não interferisse significativamente no movimento dos dedos.

O número de espiras das bobinas foi determinado com base nas restrições de dimensão. Diversos modelos de bobinas foram fabricados sendo que o número de espiras que mais se aproximou das especificações de dimensão foi de 75 espiras. Ambas as bobinas dos sensores e dos geradores foram fabricadas com esse número de espiras, com exceção da bobina localizada na palma da luva. Para essa bobina, foi definida que suas dimensões fossem tal que o diâmetro bobina pudesse abranger todos os sensores das extremidades dos dedos quando totalmente flexionados. O Quadro 6 apresenta as características dos circuitos.

**Quadro 6 - Características físicas e elétricas dos circuitos**

Circuitos sensores e geradores	Diâmetro Interno (mm)	Diâmetro Externo (mm)	Número de Espiras	Indutância ( $\mu$ H)	Capacitância Aproximada (nF)
Palma	50	55	166	2443	1
Demais circuitos	7	14	75	86-91	33

Embora as bobinas tenham sido confeccionadas com o mesmo número de espiras, devido ao processo manual de enrolamento, as dimensões das bobinas apresentaram pequenas variações entre si. Essas variações resultaram em certa diferença de indutância entre as bobinas produzidas. Os capacitores utilizados também apresentam certo erro do valor de suas capacitâncias. Assim, para que a ressonância dos circuitos seja atingida, a frequência do sinal de alimentação dos geradores terá que ser ajustável para obter a amplitude máxima de saída para cada par de circuitos.

Para que os circuitos pudessem ser fixados na luva fez-se uso de uma proteção plástica abrangendo toda a área da bobina e do capacitor. A proteção conta com diversos furos no centro para possibilitar a costura os circuitos no tecido da luva. A Figura 28 ilustra o aspecto final dos circuitos dos sensores.

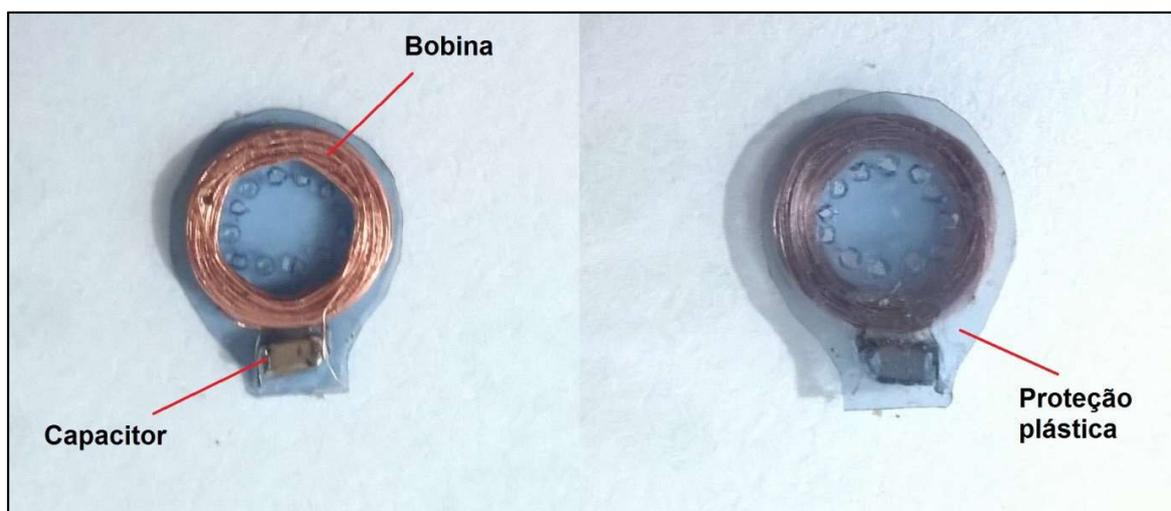


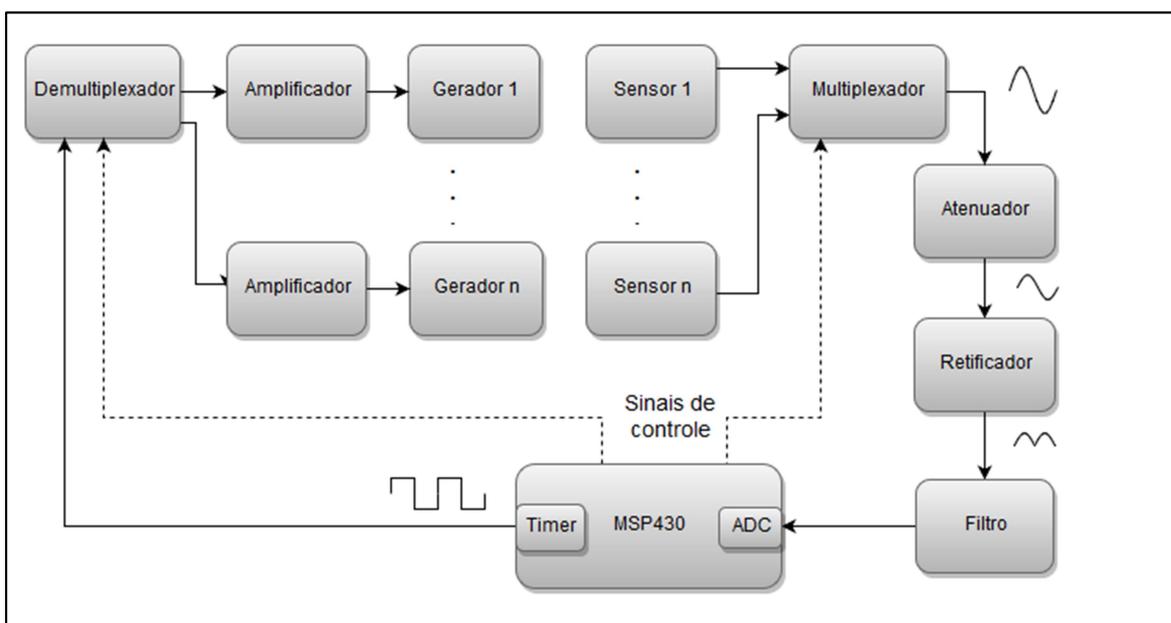
Figura 28 - Sensor indutivo desenvolvido

#### 4.3 CIRCUITO DE CONDICIONAMENTO DE SINAIS

Com o conjunto de sensores definidos, foi realizado o projeto do circuito de condicionamento de sinais. Uma das funcionalidades do circuito é possibilitar que cada gerador e cada sensor possam ser ativados individualmente, a fim de diminuir a interferência entre os sinais. Outra funcionalidade é o ajustar o ganho de tensão de resposta dos sensores e condicionar o sinal de resposta dos sensores para que

estes apresentem um sinal de CC com tensão dentro dos valores suportados pelo ADC do MSP430F5529.

A modelagem do circuito é apresentada no diagrama da Figura 29. A modelagem proposta por Fahn e Sun (2005) foi adaptada de maneira a simplificar o circuito. Uma modificação foi a eliminação do circuito de geração de onda senoidal para a alimentação dos geradores. No lugar, foi decidido utilizar um sinal de onda quadrada para esta função, pois esse tipo de sinal pode ser gerado facilmente através do microcontrolador.



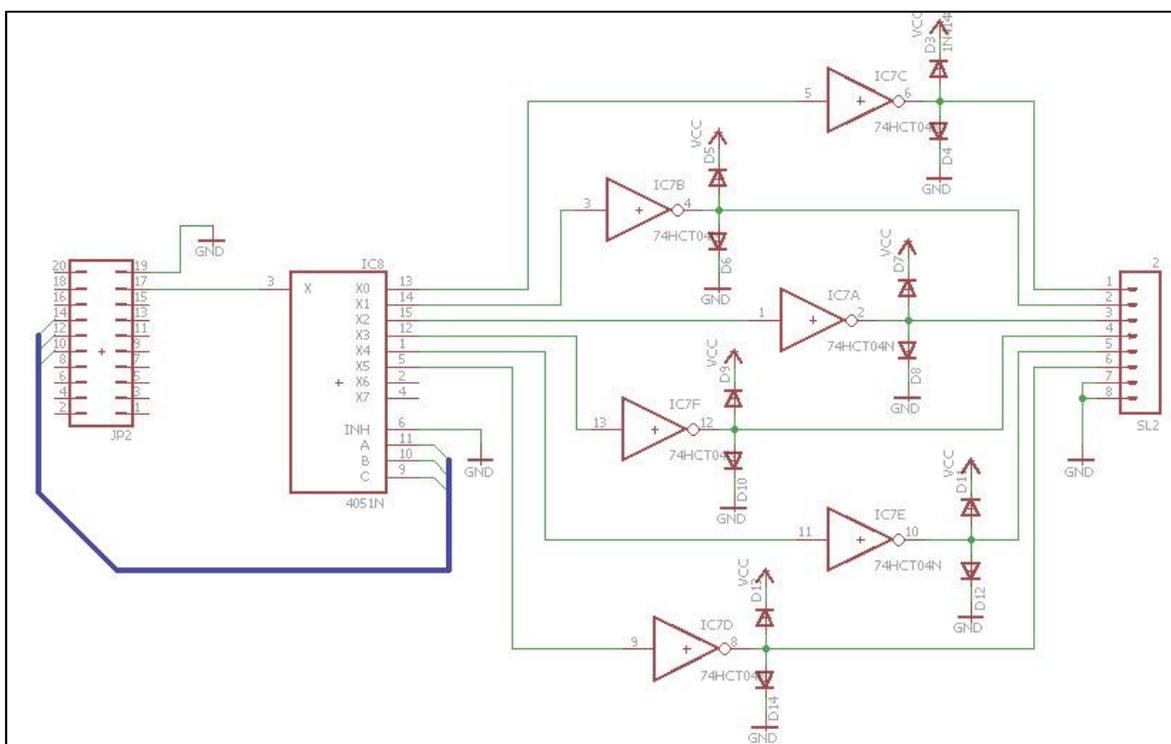
**Figura 29 - Diagrama do circuito de condicionamento de sinais.**

A mudança de forma de onda de alimentação dos geradores não interfere significativamente na resposta dos sensores, uma vez que é a frequência da onda que determina a ressonância dos geradores. O sinal de alimentação dos geradores é enviado até a unidade de demultiplexação do sinal dos geradores que é implementada pelo multiplexador analógico CD4051. Esse circuito integrado é capaz de multiplexar até oito canais, sendo suficientes para chavear todos os seis circuitos geradores.

Devido às limitações de corrente existente para as portas do microcontrolador, se faz necessário que o sinal de excitação receba um ganho de corrente. Assim, no projeto do circuito, cada canal de saída do multiplexador foi conectado a uma porta inversora, cujo funcionamento se assemelha a um

amplificador inversor. A porta inversora utilizada foi um circuito integrado 74HC04, que possui um encapsulamento com seis portas inversoras, sendo suficiente para auxiliar no fornecimento de corrente para os geradores.

Os geradores, quando conectados e alimentados, causam distorção da onda quadrada de excitação na saída da porta inversora. Para evitar que esses picos danifiquem o circuito, se fez necessário adicionar diodos na saída da porta com a função de cortar qualquer tensão que ultrapasse a tensão de alimentação da porta ou que a onda distorça até produzir corrente negativa. No diagrama da Figura 30 é ilustrada a topologia do circuito envolvido na alimentação dos geradores.



**Figura 30 - Circuito de alimentação dos geradores.**

Para trabalhar o condicionamento dos sinais de resposta dos sensores dentro dos limites de tensão de alimentação do circuito, foi utilizada uma tensão de referência, equivalente a metade da tensão de alimentação dos circuitos com o circuito divisor de tensão da Figura 31. Em suma, este circuito aplica um nível CC (*offset*) ao sinal para que este assumas tensões positivas apenas.

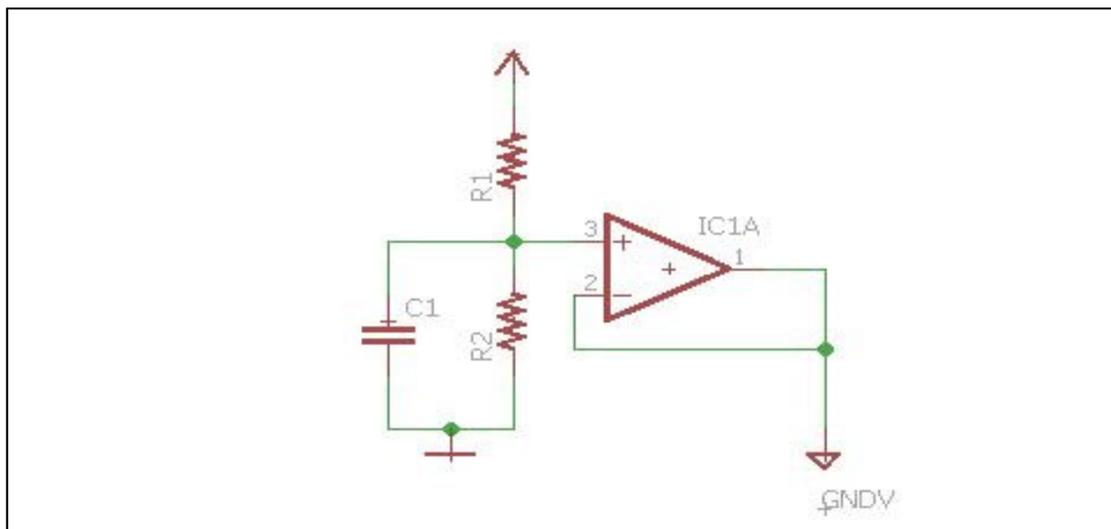


Figura 31 - Circuito divisor de tensão.

Os circuitos dos sensores são multiplexados com a topologia apresentada na Figura 32. A multiplexação é realizada através das chaves analógicas do CI DG411. Este CI foi utilizado por apresentar menor impedância de chave fechada e maior isolamento de ruídos entre os canais, algo que foi evidenciado no CI CD4051.

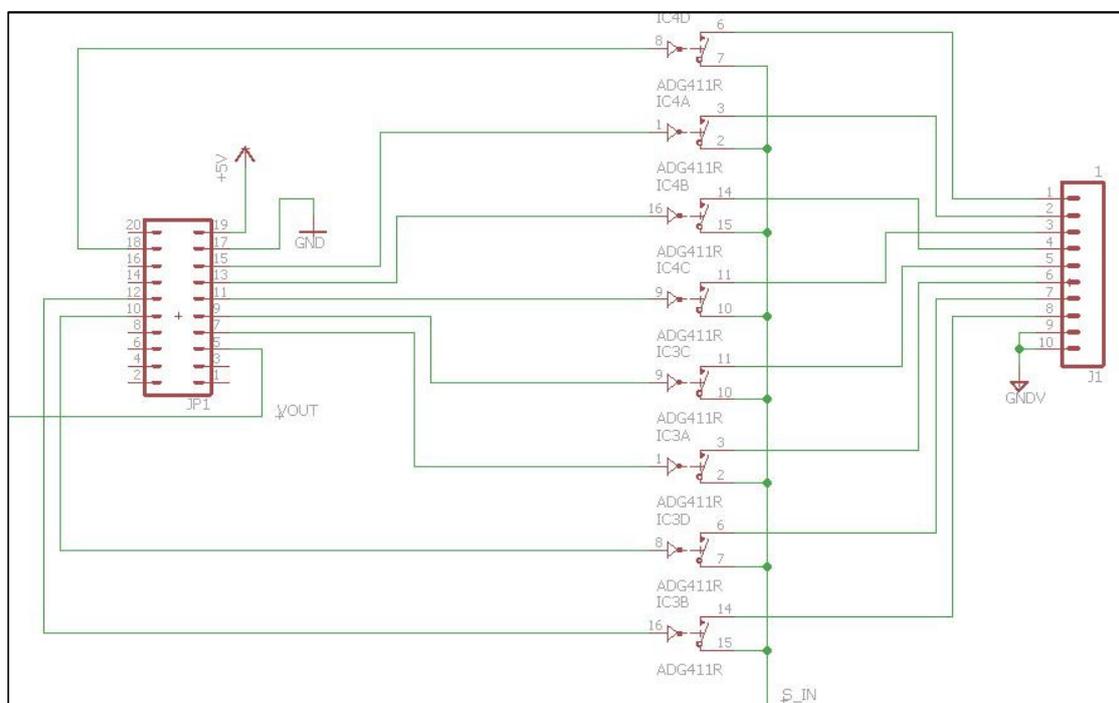
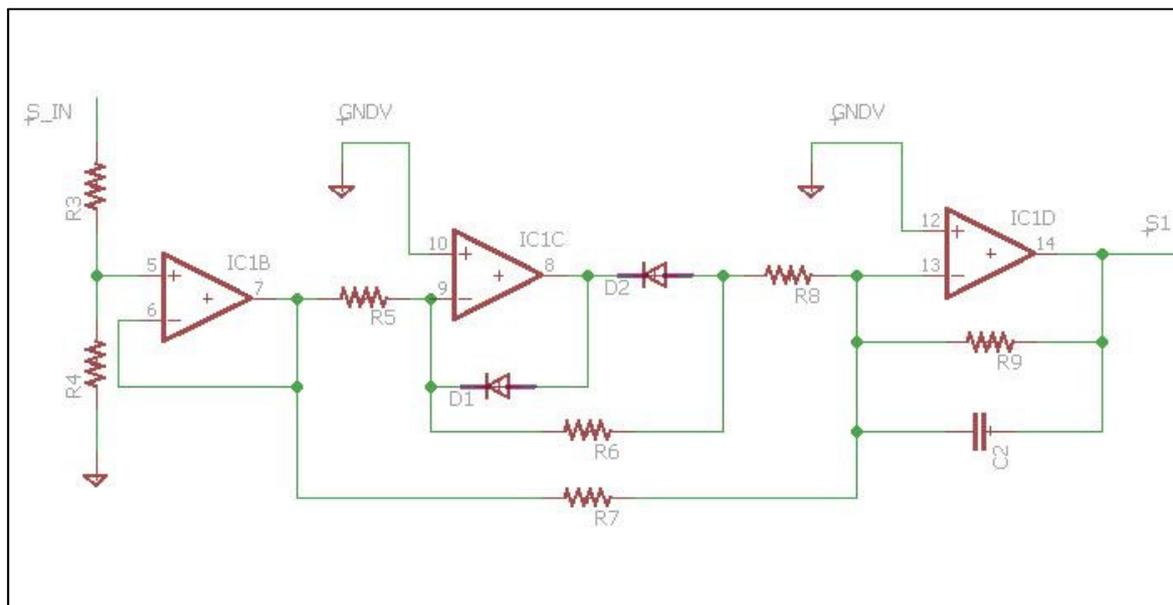


Figura 32 - Circuito de multiplexação dos sensores.

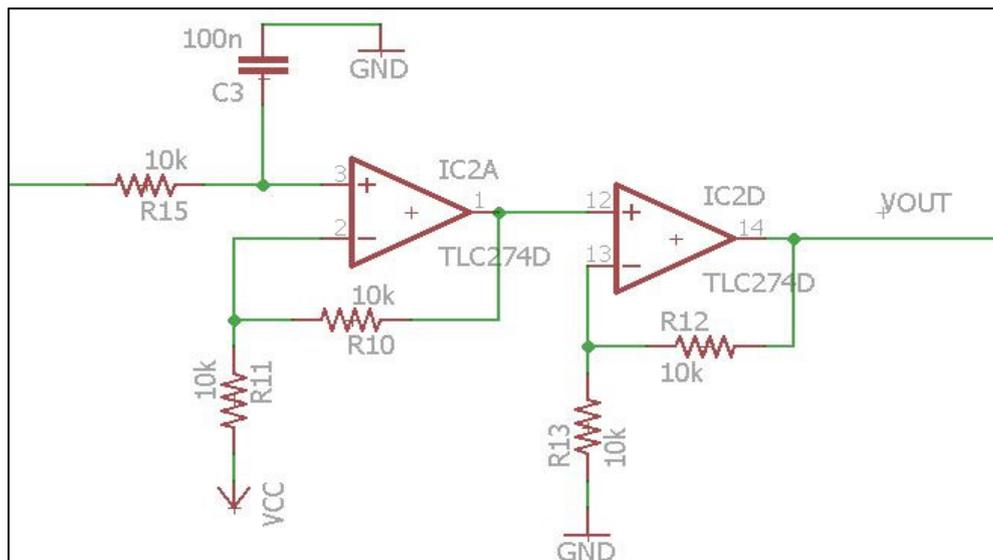
De acordo com a Figura 33 a topologia conecta as portas de controle das chaves ao barramento de pinos da placa do microcontrolador. Todas as chaves tem um de seus terminais conectados em comum (representado do diagrama por S\_IN) e o outro terminal conectado em um pino de outro barramento onde é conectado o cabo da luva. O software de controle de leitura dos sensores precisará garantir que apenas uma chave esteja fechada por vez para a multiplexação seja efetuada.



**Figura 33 - Circuito de retificação**

O sinal multiplexado (S\_IN) passa por um divisor de tensão para que o sinal seja atenuado, evitando a saturação da saída dos circuitos seguintes. Para obter a tensão de amplitude do sinal de entrada fez-se uso do circuito de retificação apresentado na Seção 3.2.2. A escolha desse retificador se deve ao fato de ser mais preciso em relação a outras topologias.

Na saída do circuito de retificação foi adicionado um filtro para transformar o sinal oscilatório em uma tensão de corrente contínua. Nesse estágio do circuito também é necessário que o sinal assumo o mesmo referencial de tensão do microcontrolador e amplificar o sinal para compensar a atenuação anterior. Na Figura 34 é mostrado o circuito de pós-retificação modelado.



**Figura 34 - Circuito pós-retificação.**

No primeiro AmpOp, o divisor de tensão dos resistores R11 e R10 garante que o sinal de saída assumo o referencial de alimentação do AmOp. O segundo AmpOp adiciona um fator de ganho ao sinal. A tensão de saída do circuito (VOUT) é conectada até o pino de entrada do conversor digital analógico do microcontrolador.

O circuito de condicionamento conta ainda com barramentos de pinos para que se possa encaixar a placa de desenvolvimento do microcontrolador e o cabo de conexão com a luva. Os barramentos garantem maior praticidade para programar e testar o sistema uma vez que este poderá ser conectado e desconectado sempre que for necessário.

A fonte de alimentação do circuito foi definida como sendo a mesma alimentação da placa de desenvolvimento do microcontrolador, que é alimentado através de um cabo do tipo USB (do inglês, Universal Serial Bus – Barramento Serial Universal). Dessa maneira, o sistema pode funcionar apenas com a tensão fornecida por qualquer computador com uma porta serial. A tensão da porta serial (5V) é regulada para uma tensão adequada para alimentar o microcontrolador (VCC = 3.3V). Essa regulação é realizada através do circuito da Figura 35, que também possui um diodo emissor de luz para identificação visual da energização da placa.

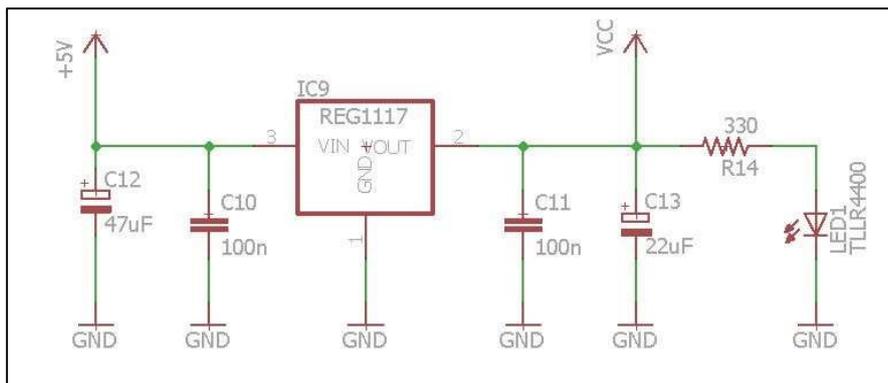


Figura 35 - Circuito de regulação de tensão

#### 4.4 MODELAGEM DO *FIRMWARE*

O *firmware* foi modelado de acordo com o fluxograma da Figura 36. As principais funções em quatro partes, inicialização das portas e periféricos do microcontrolador que serão utilizados no sistema, varredura de ativação dos geradores e leitura dos sensores, a classificação dos padrões de dados de entrada através de uma rede neural artificial, e por fim a exibição do caractere da letra sinalizada pelo usuário.

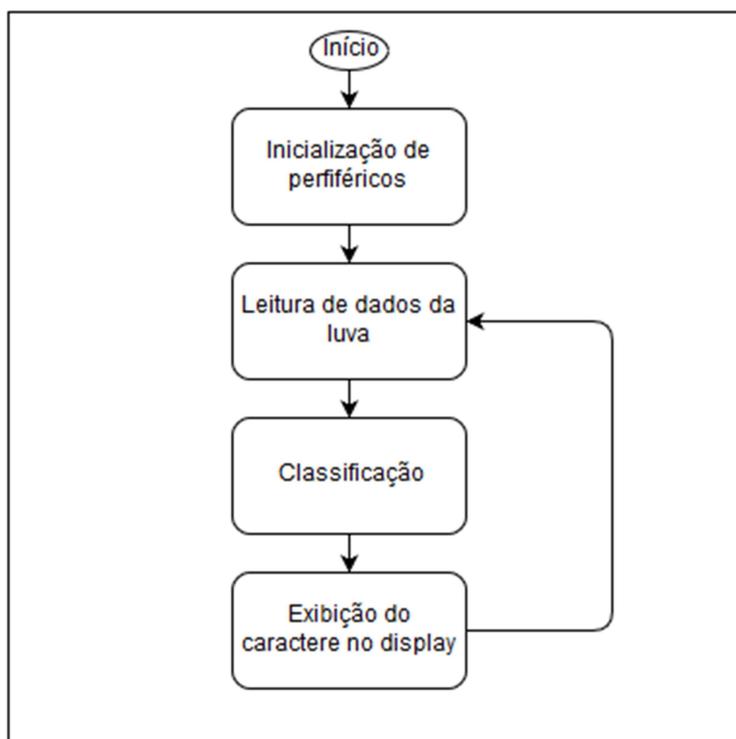


Figura 36 - Fluxograma principal do sistema.

#### 4.4.1 Inicialização das Portas e Periféricos

Quando se inicia o sistema, os periféricos do microcontrolador serão inicializados de acordo com as especificações do *datasheet* atendendo os requisitos de funcionamento. Os periféricos utilizados para o sistema são as portas de entrada e saída para controlar o circuito de chaveamento, um conversor analógico-digital para leitura do nível de tensão de resposta do sensor selecionado, um temporizador para geração do sinal de onda quadrada e uma unidade de comunicação síncrona e assíncrona para comunicação do microcontrolador com o módulo de *display*.

O microcontrolador MSP430F5529 possui 8 portas de entrada e saída com oito bits cada. Para controlar o circuito de chaveamento dos geradores são necessários apenas três bits para endereçar os seis canais dos geradores. O controle do chaveamento dos sensores se dará com um bit para cada sensor, ou seja, oito bits são necessários. Assim, apenas duas portas de entrada e saída seriam suficientes para o controle do chaveamento. A escolha das portas também se justificou para facilitar o roteamento da placa de circuito impresso, minimizando a distância entre as portas do microcontrolador e as portas dos CIs responsáveis pelo chaveamento.

O temporizador será inicialmente configurado para trabalhar com o clock principal do microcontrolador. A configuração do temporizador para que se produza uma onda quadrada se dará configurando o temporizador no modo de modulação de largura de pulso, ou PWM (*Pulse Width Modulation*), definindo o ciclo de maneira que a onda possua uma frequência de 100 KHz e uma largura de pulso equivalente a metade do período (razão cíclica de 50%).

O conversor analógico digital é configurado para utilizar o clock principal, no modo de conversão simples. A interrupção associada à conversão tem que estar ativa para que se possa ter controle de quando a leitura do sensor foi realizada. Assim, cada vez que for solicitada uma conversão, uma variável é atualizada dentro da rotina de interrupção, atualizando o valor digital convertido.

#### 4.4.2 Varredura dos Sensores e Geradores

A aquisição dos dados da luva se dá através de uma rotina responsável pelo controle do circuito de multiplexagem e leitura do sinal de saída do circuito de condicionamento. A rotina realiza a seleção dos canais dos geradores e sensores de maneira que apenas um par de circuitos seja ativo por vez. Essa técnica de multiplexação por tempo reduz a interferência que o campo gerado pelos geradores interfira na resposta dos sensores.

Considerando as limitações físicas do movimento da mão humana e a distância de alcance mínima dos geradores, a varredura se limita à combinação de geradores e sensores possíveis de se realizar e que formam alguma configuração de mão do alfabeto da Libras. No Quadro 7 são exibidas as relações entre os geradores e os sensores de alcance e interesse para a aplicação, excluído a relação do gerador G3 e do sensor S4 devido ao fato de que a leitura desse par de ativação resultou em zero para todas as amostras de treinamento.

**Quadro 7 - Pares de ativação.**

<b>Iteração</b>	1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Gerador</b>	G1	G1	G1	G1	G1	G2	G3	G3	G3	G3	G4	G5	G6
<b>Sensor</b>	S1	S2	S3	S4	S5	S5	S1	S2	S3	S6	S6	S7	S8

O Quadro 7 representa uma generalização do mapeamento realizado na Seção 4.2.1, porém, com a exclusão da relação entre o gerador G3 com o sensor S4 devido ao fato de que durante a etapa de aquisição de dados para as configurações de mão não obteve-se nenhum valor diferente de zero para este par para.

Desta forma a varredura consiste em um laço com treze iterações, uma para cada par do Quadro 7. Ao fim de cada iteração do laço de varredura é solicitada uma leitura do ADC e um valor de doze bits correspondente à tensão do sensor naquele instante é obtido e armazenado em um vetor de dados.

#### 4.4.3 Reconhecimento de Padrões

A terceira parte do software é o reconhecimento de padrões utilizando uma rede neural artificial. A rede utilizada é do tipo multicamada, com três camadas

no total (uma de entrada, uma de saída e uma camada oculta). A modelagem da rede buscou utilizar uma arquitetura suficiente para a resolução do problema de classificação.

A camada de entrada foi definida com um neurônio para cada varredura necessária da luva, ou seja, treze neurônios. Na camada de saída constará o número de neurônios correspondente ao número de configurações de mão que o conjunto de sensores é capaz de identificar.

Neste caso, como a luva não contará com um sensor para definir o movimento e aceleração da mão, os sinais das letras que possuem configuração de mão idêntica não serão reconhecidos pelo classificador. Dessa forma, os sinais das letras H, K e J são desconsiderados para o projeto da rede.

A única camada oculta da rede foi definida através do desempenho do treinamento de diferentes números de neurônios, visando identificar o número mínimo de neurônios para que a rede apresente a melhor resposta diante do conjunto de dados de treinamento. No Quadro 8 são apresentadas as topologias candidatas para a aplicação.

**Quadro 8 - Topologia de redes candidatas**

Rede Candidata	Número de neurônios da camada oculta
Rede 1	19
Rede 2	21
Rede 3	23
Rede 4	25

#### 4.4.4 Simulação da Rede Neural

Os parâmetros de treinamento utilizados para as topologias candidatas foram os parâmetros padrões da função de treinamento do MATLAB para redes alimentadas adiante. As funções de ativação dos neurônios e o algoritmo de treinamento também utilizaram das especificações padrão, sendo que a função de ativação é a função tangente hiperbólica e o algoritmo é o *backpropagation* Levenberg-Marquardt. A parada do treinamento é feita por validação cruzada,

utilizando divisão do conjunto de dados de treinamento com 70% das amostras para treinamento, 15% para validação e 15% para teste, selecionadas aleatoriamente.

O conjunto de treinamento foi desenvolvido obtendo-se 25 amostras de entradas de dados da luva para cada classe de padrões identificáveis. As classes, como foram citadas anteriormente, constituem as letras do alfabeto com exceção das letras J, H, K.

Durante a realização da amostragem observou-se que os padrões da letra Z apresentaram semelhança considerável com os padrões da letra X. Optou-se então em remover esta última do conjunto de classes identificáveis. Uma classe adicional se fez necessária para identificação de uma configuração de mão de descanso, uma vez que, neste momento, não será implementado um algoritmo que identifique quando uma configuração está sendo realizada pelo usuário.

Assim, os padrões de treinamento formam uma matriz de dimensões com 575 linhas (25 amostras de 23 classes) por 13 colunas (13 dados de entrada). Em conjunto com a matriz de padrões, para o treinamento da rede é necessário uma matriz que associe os padrões da matriz P com as classes de saída da rede, também chamada de matriz de padrões alvos, denominada T, com dimensões 575x23.

O *script* utilizado para simular a rede neural no MATLAB é mostrado na Listagem 1. O script se inicia com a chamada da função “CarregaDados”, que inicia as matrizes P e T com os dados do treinamento. Em seguida é criada uma rede neural com a função “feedforwardnet”. O parâmetro desta função é o número de neurônios utilizados, variando de acordo com as topologias treinadas.

```
[P, T] = CarregaDados(); %carrega dados de treinamento
neuroniosOcultos = 23; %número de neurônios da camada oculta

% cria uma rede neural do tipo feedforward
net = feedforwardnet(neuroniosOcultos);

%treina rede
[net, tr, Y, E] = train(net, P, T);

[c, cmind, per] = confusion(T, Y);
```

**Listagem 1 - Roteiro de criação e treinamento da rede neural.**

A função “train” é responsável por treinar a rede com os dados dos parâmetros. A função abre na tela uma janela onde é possível visualizar diferentes aspectos do treinamento, como a topologia da rede, os critérios de paradas e plotagem de gráficos de treinamento.

Um aspecto da função de treinamento do MATLAB é que a mesma realiza um pré-processamento dos dados de treinamento. Esse processamento inclui a eliminação de linhas nulas das matrizes e mapeamento dos dados para valores no intervalo [-1, 1]. Assim, na implementação do código no microcontrolador esse processamento precisou ser replicado para o correto funcionamento da rede.

Por fim, é chamada a função “confusion” que retorna diversas informações sobre o desempenho da rede. A variável ‘c’, por exemplo, apresenta a percentagem de padrões que não foram devidamente classificados no treinamento. O *script* foi executado para as topologias candidatas para verificar a topologia que apresentasse o melhor desempenho.

## 4.5 IMPLEMENTAÇÃO

A seguir são descritos os passos da implementação dos códigos de aquisição, processamento de dados e interface do sistema. A implementação se iniciou com a rotina de aquisição dos dados dos sensores da luva. Em seguida, foi implementado o processo de classificação simulado.

### 4.5.1 Aquisição dos Dados da Luva

A aquisição dos dados da luva é realizada através da rotina “RecebeDadoSensor”, cujo código é exibido na Listagem 2. Essa rotina realiza a varredura dos sensores e geradores da luva de acordo com a relação de geradores e sensores definidos de acordo com Quadro 7.

A rotina consiste em um laço com treze iterações, referente às relações do quadro. O laço se inicia selecionando a frequência de ressonância ideal para a iteração através da função “\_SelecionaFrequencia”. Em seguida, a função “\_SelecionaGerador” seleciona os bits de controle do demultiplexador para selecionar o canal do gerador.

```

Void RecebeDadosSensores() {
    PORTMUX &= ~(BIT2 + BIT3 + BIT4);
    unsigned int i = 0;
    while (i < iteracoesVarredura) {
        _SelecionaFrequencia(i);
        _SelecionaGerador(i);
        __delay_cycles(75000);
        _SelecionaSensor(i);
        __delay_cycles(75000);

        ADC12CTL0 |= ADC12ENC + ADC12SC;
        while (ADC12CTL0 & ADC12BUSY)
            __bis_SR_register(CPUOFF + GIE);

        vtEntrada[i] = normaliza(((float) (vlrADC) / 4095.0), i, NORMALIZA);
        i++;

        PORTCHAN |= 0xff;
        P1OUT |= BIT6;
    }
}

```

**Listagem 2 - Rotina de varredura dos sensores.**

Após a seleção do canal, é necessário certo intervalo de tempo para que o gerador tenha tempo de entrar em ressonância. Esse tempo é estabelecido através da função `__delay_cycles()`, onde o microcontrolador permanece ocioso pelo número de ciclos de clock estipulado. O tempo observado para a estabilização do gerador foi de seis milissegundos.

Em seguida, é selecionado o canal do sensor de interesse, também aguardando um tempo para que a resposta do sensor se estabilize. A seleção se dá através da seleção do pino do microcontrolador em nível baixo. Após a seleção do gerador e do sensor, o próximo passo da rotina é a leitura do ADC. Após o término da conversão, o valor convertido é normalizado através da função “normaliza” e armazenado no vetor de entrada do sistema de classificação. As funções de seleção e de normalização que aparecem na Listagem são disponibilizadas no Apêndice A.

#### 4.5.2 Implementação da Rede Neural

O código da rede neural implementado no microcontrolador foi desenvolvido iniciando-se pela representação dos neurônios. Para isso, fez-se uso de estruturas do tipo vetor para armazenar os dados dos neurônios, como é mostrado na Listagem 3.

```

float static vtEntrada[NRO_ENTRADA];
float vtSaida[NRO_SAI DA];

const float vtBiasO[NRO_OCULTA] = { ... };
const float vtBiasS[NRO_SAI DA] = { ... };

const float vtPesoEO[NRO_ENTRADA][NRO_OCULTA] = { ... };
const float vtPesoOS[NRO_ENTRADA][NRO_OCULTA] = { ... };

```

**Listagem 3 - Representação dos neurônios.**

Os vetores “vtEntrada” e “vtSaida”, correspondem aos valores de ativação dos neurônios das camadas de entrada e saída, respectivamente. Os vetores “vtBiasS” e “vtBiasO”, representam os valores limiares dos neurônios de saída e ocultos, respectivamente. Os vetores “vtPesoEO” e “vtPesoSO” são responsáveis em armazenar os pesos sinápticos das conexões entre as camadas de entrada e oculta, e das conexões entre as camadas oculta e de saída, respectivamente. Os vetores de limiar e de pesos sinápticos são inicializados de acordo com os valores obtidos na simulação, omitidos na listagem para facilitar a visualização.

A rotina “RNA”, exibida na Listagem 4, propaga os dados de entrada através da rede e atualiza o vetor “vtSaida”. A rotina se divide em duas etapas: na primeira parte é realizada uma varredura na matriz “vtPesoEO”, onde para cada neurônio da camada oculta é calculado a soma dos valores de entrada ponderados.

```

void RNA() {

    float vtOculta[NRO_OCULTA];
    unsigned int i, j, k;
    float soma = 0;

    for (j = 0; j < NRO_OCULTA; j++) {
        soma = (vtBiasO[j]);
        for (i = 0; i < NRO_ENTRADA; i++) {
            soma += (vtEntrada[i] * vtPesoEO[i][j]);
        }
        vtOculta[j] = (tanh(soma));
        soma = 0;
    }
    float vtSomaS[NRO_SAI DA];
    for (k = 0; k < NRO_SAI DA; k++) {
        vtSomaS[k] = (vtBiasS[k]);
        for (j = 0; j < NRO_OCULTA; j++) {
            vtSomaS[k] += (vtOculta[j] * vtPesoOS[k][j]);
        }
        vtSaida[k] = normaliza((vtSomaS[k]), i, DESNORMALIZA);
    }
}

```

**Listagem 4 - Rotina RNA.**

A variável de soma é iniciada com o limiar do neurônio. Depois de somados os pesos, a função “FuncaoAtivação” calcula o valor de ativação do neurônio da camada oculta.

A segunda parte é análoga à primeira, porém agora os valores de ativação dos neurônios da camada oculta são usados para calcular a saída. Portanto, é realizada a varredura do vetor “vtPesoOS” somando os valores recém calculados dos neurônios ocultos multiplicados pelo peso correspondente do vetor.

A soma para cada neurônio de saída retornado é mapeado pela função “normaliza” para assumir um valor entre zero e um. Ao final da rotina o vetor “vtSaida” é atualizado com o resultado da classificação da rede neural.

#### 4.5.3 Interpretação da Saída da Rede Neural

Após o vetor de dados de saída ser atualizado, se fez necessário associar o resultado com a interpretação do mundo real para a qual a rede foi proposta através da função “PosProcessamentoRNA”, exibida na Listagem 5. A função consiste em uma varredura no vetor de saída para localizar maior valor. O índice do vetor que contém o maior valor é retornado.

```

unsigned int PosprocessamentoRNA() {
    float maxVlr = 0;
    int maxi = 0;
    int i;
    for (i = 0; i < NRO_SAI DA; i++) {
        if (vt Sai da[i] > maxVlr) {
            maxVlr = vt Sai da[i];
            maxi = i;
        }
    }
    return maxi;
}

```

**Listagem 5 - Função de pós processamento.**

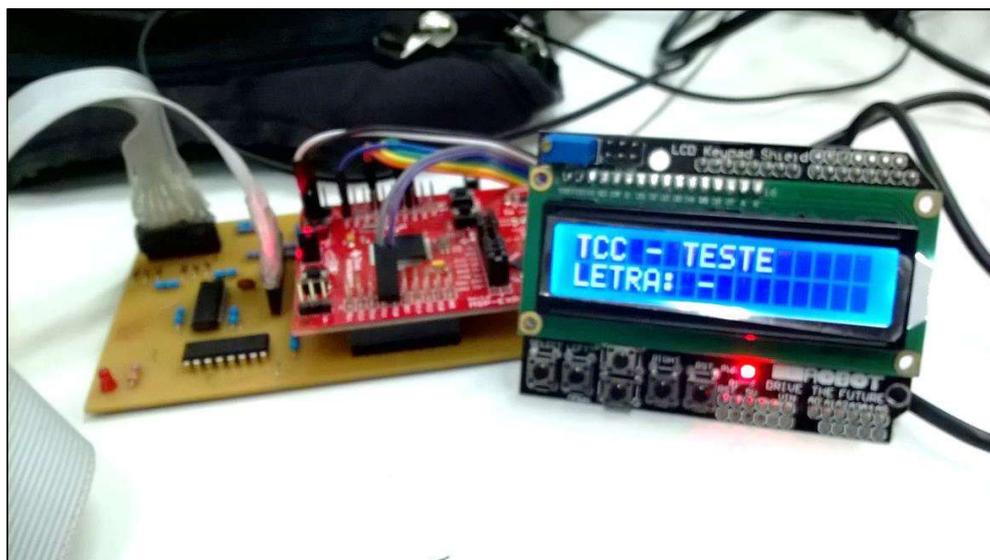
O índice retornado corresponde à classe que foi reconhecida pela rede. Esse índice é utilizado no *loop* principal para selecionar em um vetor de caracteres, o caractere que corresponde à classe identificada. O display é atualizado, imprimindo o caractere selecionado.

## 5 RESULTADOS E DISCUSSÕES

A seguir são apresentados o aspecto final do sistema e os resultados dos testes realizados durante o desenvolvimento do sistema. Os testes serviram para validar as modelagens de cada etapa e determinar o desempenho do sistema dentro do escopo proposto.

### 5.1 ASPECTO FÍSICO DO SISTEMA

A seguir são apresentadas algumas fotografias que ilustram o aspecto do sistema desenvolvido. A Figura 37 mostra, à esquerda, a placa de circuito eletrônico de condicionamento de sinais com a placa de desenvolvimento do microcontrolador devidamente conectada. À direita, encontra-se o módulo LCD exibindo o texto definido para os testes.



**Figura 37 - Circuito de condicionamento, microcontrolador e display.**

A luva desenvolvida é exibida na Figura 38, onde, à esquerda tem-se uma visão da luva mostrando a palma e os dedos esticados. Nesta visão é possível visualizar os sensores das extremidades dos dedos indicador, médio, anelar e mínimo, assim como os geradores da palma e da extremidade do polegar.

À direita da Figura 38 é apresentada uma visão lateral da luva, podendo ser visualizado os sensores e geradores localizados nas laterais dos dedos e da

palma. Também é possível observar como os fios que conectam os circuitos de sensoriamento foram costurados na superfície da luva, tendo como objetivo reduzir a tração nos fios quando os dedos são flexionados.



**Figura 38 - Vista frontal e lateral da luva sensora desenvolvida.**

O sistema completo, com todos os componentes conectados é exibido na Figura 39. Nesta figura pode-se observar o conector fixado nas costas da luva, com intuito de possibilitar o encaixe do cabo de conexão da luva com a placa desenvolvida.



**Figura 39 - Sistema completo e conectado.**

## 5.2 RESPOSTA DOS SENSORES

Um dos testes do sistema consistiu em avaliar a resposta dos sensores em relação à distância e ângulo dos geradores para diferentes. A avaliação buscou conferir a sensibilidade do sensor. O teste de distância foi realizado em relação ao sensor S1 em relação à distância do gerador G1. No teste de angulação foi realizado com o sensor S6 e G4.

**Tabela 1 - Resposta do sensor S1 em relação à distância do gerador G1.**

Distância aproximada (mm)	Tensão de pico aproximada (mV)
0,0	3,5
0,5	3,3
1,0	2,9
1,5	2,5
2,0	2,2
2,5	2,1
3,0	2,0
4,0	1,8
4,5	1,7
5,0	1,7

**Tabela 2 - Resposta do sensor S6 em relação ao ângulo entre o gerador G4.**

Ângulo aproximado (°)	Tensão de pico aproximada (V)
0	4,4
10	3,2
15	2,8
20	2,6
30	2,3

Os resultados apresentados nas tabelas permitem identificar a significativa variação de tensão em relação às distâncias e ângulos em relação aos geradores. O sensor S1 em comparação com o sensor S6 demonstrou possuir uma resposta inferior quando a proximidade em relação ao gerador é máxima. Isso se deve ao fato do melhor acoplamento que existe entre o sensor S6 e o gerador G4 por estas possuírem o mesmo formato e número de espiras. No caso do sensor S1, o gerador G1 possui uma área muito maior, fazendo com que a transferência de energia para o sensor S1 ser dada parcialmente. Quando outros sensores são aproximados do gerador G1, observou-se que a resposta de S1 fica ainda menor na proximidade máxima, obtendo-se aproximadamente 2 V. Afastando o sensor S1 em

1 cm, aproximadamente, a tensão apresenta tensão próxima da tensão de referência (1,7 V).

Isso resulta que, embora as dimensões escolhidas do gerador G1 possibilitarem a simplificação do sensoriamento (uma vez que elimina a necessidade de haver na palma da luva um gerador para cada sensor das extremidades dos dedos), a escolha acarretou em uma limitação da distância mínima de resposta dos sensores. Dessa forma, o padrão de dados de entrada para algumas configurações de mão acabaram por não se enquadrar no mapeamento idealizado na seção 4.2.1. A configuração da letra X não fez parte do conjunto de classes identificáveis, pois não foi possível detectar a flexão parcial do dedo indicador pelos sensores.

### 5.3 POSICIONAMENTO DOS SENSORES NA LUVA

Apesar de todos os cuidados no requisito da luva não interferir na execução das configurações de mão, durante a fase de testes notou-se que o atrito entre os circuitos localizados nas laterais dos dedos resultou em uma dificuldade na realização de alguns sinais. Por exemplo, na sinalização da letra R é necessário um esforço adicional para posicionar o dedo indicador sobre o dedo médio.

A bobina do gerador da palma da luva dificulta o movimento de sinais que exigem que o polegar precise se deslocar sobre a palma, como é o caso da letra B. Em sinais de letras que exigem que alguns dedos fiquem flexionados e outros esticados, como o caso da letra U e W, a bobina do sensor sofre flexão por causa da deformidade que a superfície da palma apresenta nessas configurações de mão. Essa flexão é prejudicial para a integridade do sensor, podendo ocasionar a ruptura dos fios da bobina.

O posicionamento dos circuitos na luva teve que ser realizado diversas vezes até que a posição ideal para o máximo acoplamento fosse encontrada, principalmente para o sensoriamento das laterais dos dedos. A diferença de alguns milímetros resulta em uma variação significativa da resposta do sensor em determinados sinais. Por exemplo, nos sinais das letras U e D, observou-se um desalinhamento entre o sensor S8 e o gerador G6 que resulta em uma resposta do sensor ligeiramente menor do que a esperada.

Apesar das dificuldades expostas, a modelagem do sensoriamento possibilitou um conjunto de dados suficiente para estabelecer padrões de resposta bem definidos quando as configurações de mão são executadas dentro do previsto. Com exceção da letra X, devido aos fatores anteriormente citados, a coleta de amostras de dados para o treinamento do sistema de classificação resultou que até mesmo sinais que poderiam apresentar dificuldade de reconhecimento, como os sinais das letras G e Q, apresentaram um padrão de dados distintos que permitiram diferenciá-los e conseqüentemente, identificá-los.

Dessa forma o modelo do posicionamento dos sensores garantiu que um mínimo de sensores indutivos fosse utilizado para a caracterização dos dados do posicionamento dos dedos. Com uma modelagem reformulada, adicionando um número maior de sensores é possível aplicar o modelo matemático sugerido por Fahn e Sun para determinar, aproximadamente, o ângulo entre cada segmento dos dedos.

#### 5.4 TESTE DA CLASSIFICAÇÃO

A seguir são apresentados os resultados das simulações das diferentes topologias de redes neurais. A Tabela 3 apresenta alguns dados de treinamentos das diferentes topologias.

**Tabela 3 - Resultado dos treinamentos das topologias candidatas.**

<b>Rede candidata</b>	<b>Rede 1</b>	<b>Rede 2</b>	<b>Rede 3</b>	<b>Rede 4</b>	<b>Rede 5</b>
<b>Número de neurônios ocultos</b>	13	19	21	23	26
<b>Erro médio quadrático</b>	0,183	0,108	0,00535	0,00435	0,00459
<b>Percentagem de acerto (%)</b>	73,04	92,17	93,57	97,22	97,21
<b>Época erro mínimo</b>	34	17	19	28	27



Como a resposta do sensor varia rapidamente com o aumento da distância entre o gerador, uma variação da maneira como a configuração de mão é sinalizada pode gerar uma classificação errônea. Por exemplo, na configuração da letra B, se o usuário posicionar o polegar de maneira que o sensor S5 fique fora do alcance do gerador G2, irá resultar em um vetor de dados semelhante à da configuração de letra C. O conjunto de sensores também influencia o resultado da classificação das configurações de mão, uma vez que o modelo definido de sensoriamento apresenta diversos pontos cegos.

Uma solução seria remodelar o conjunto de sensores para minimizar os pontos cegos. Acredita-se que adicionando mais sensores e geradores sobre a luva é possível reduzir erros de classificação de configurações e até mesmo possibilitar que todas as configurações que compõe a Libras possam ser determinadas. Aumentar o conjunto de treinamento da rede, utilizando outros usuários para coletar as amostras, poderá melhorar a capacidade de generalização da rede neural.

Para validar a implementação da rede neural escolhida no microcontrolador foi realizado um teste de saída da rede para alguns padrões de dados. Na Tabela 5 é apresentado um dos padrões utilizados no teste, que corresponde à configuração da letra A. Na Tabela 6 é apresentada as repostas da rede simulada e implementada, assim como a saída ideal.

**Tabela 5 - Padrão de dados da letra A.**

Padrão de teste: letra A	
	0,3433
	0,4720
	0,3807
	0,2354
	0,0103
	0,7918
	0,0088
	0,0083
	0,0085
	0,0078
	0,9118
	1,0000
	0,9812

**Tabela 6 - Comparação entre as respostas das redes simulada e implementada.**

Saída da rede simulada	Saída da rede implementada	Saída ideal
1,0054	1,0054	1
0,0068	0,0067	0
0,0089	0,0090	0
0,0075	0,0076	0
-0,0223	-0,0222	0
-0,0148	-0,0150	0
-0,0037	-0,0036	0
0,0168	0,0168	0
0,0191	0,0191	0
-0,0037	-0,0037	0
-0,0026	-0,0027	0
0,0063	0,0064	0
0,0027	0,0026	0
0,0053	0,0053	0
-0,0034	-0,0035	0
-0,0199	-0,0199	0
-0,014	-0,0139	0
-0,013	-0,0131	0
0,0154	0,0154	0
0,0066	0,0068	0
0,0067	0,0067	0
-0,0009	-0,0008	0
-0,0092	-0,0093	0

Observando a Tabela 6 é possível notar que a rede implementada se aproxima de maneira satisfatória da rede simulada no MATLAB. Outros padrões testados também apresentaram uma resposta semelhante em ambos os casos.

## 6 CONSIDERAÇÕES FINAIS

O desenvolvimento do trabalho consistiu em projetar e implementar um sistema que obtivesse dados referente às configurações de mão de um usuário utilizando um dispositivo do tipo luva sensora. Os dados adquiridos serviram para que as letras do alfabeto da Libras sinalizadas pelo usuário pudessem ser interpretadas por um algoritmo de classificação de padrões.

Diante das diferentes possibilidades de configurações que a mão pode assumir na sinalização dos sinais da Libras, o escopo do sistema limitou-se a identificar apenas as configurações de mão do alfabeto. Para realizar o sensoriamento da luva, foi preciso definir um conjunto de sensores que possibilitassem extrair dados suficientes para identificar as principais características das configurações de mão, como foi apresentado na seção 4.2.1. O projeto desse conjunto também foi essencial para que o projeto e implementação do circuito de condicionamento não se tornasse muito complexo.

Apesar das simplificações na modelagem do sistema, o dispositivo desenvolvido se mostrou capaz de reconhecer a maioria das configurações de mão do alfabeto da Língua Brasileira de Sinais. Alguns sinais não puderam ser classificados em decorrência da limitação do escopo do sensoriamento, preferindo-se realizar o reconhecimento das configurações de mão. Dessa forma os sinais apenas diferenciados entre si pelo movimento não foram abordados, como é o caso das letras J, H, K e X.

O desenvolvimento deste trabalho serviu para dar viabilidade para trabalhos posteriores com a utilização do dispositivo de luva sensora utilizando sensores indutivos. Os sensores, apesar da curta distância detectável, se mostraram uma opção para o desenvolvimento de dispositivos de luva sensora, podendo ser utilizado em conjunto com outros sensores tipicamente utilizados, tais como, os sensores resistivos de flexão.

Outra vantagem na utilização de sensores indutivos está no fato de que pode ser facilmente construído, sem necessitar de processos industriais. Isso possibilita que o desenvolvedor do sensor possa manufaturá-lo de acordo com suas necessidades, podendo variar o diâmetro ou o número de espiras para obter maior ou menor precisão.

O microcontrolador utilizado também se mostrou adequado para a aplicação. Apesar das inúmeras operações matemáticas com ponto flutuante envolvidas no cálculo da saída da rede neural, o microcontrolador foi capaz de realizar o processamento em tempo hábil. Contudo, caso o sistema seja aprimorado em trabalhos futuros deverá existir um estudo para melhor aproveitar o poder de processamento do microcontrolador.

Por fim, as redes neurais se mostraram uma ferramenta útil no problema de reconhecimento de padrões de dados gerado pela luva. A arquitetura da rede e os parâmetros de treinamento mostraram-se suficientemente adequados. Entretanto, as limitações impostas na modelagem do projeto não deixam conclusivas se a eficiência máxima da rede foi atingida. A avaliação da melhor arquitetura e configuração da rede desenvolvida para o sistema exige um trabalho mais específico, que envolve diferentes heurísticas de otimizações.

## 6.1 TRABALHOS FUTUROS

Tendo em vista que a luva desenvolvida não apresenta sensores de movimento, os sinais que possuem esse parâmetro não puderam ser reconhecidos. Isso deixa em aberto o desenvolvimento do sistema para abranger que toda as letras do alfabeto da Libras. Para isso, se fará necessário adicionar um sensor do tipo acelerômetro ou giroscópio ao conjunto de sensores da luva.

O reconhecimento do movimento como parâmetro exige que se estabeleça um método de segmentação para determinar a duração do movimento. Esse método pode ser aprimorado para a detecção da mudança da configuração ao longo do tempo de execução do sinal e possibilitar o sistema de reconhecer sinais mais específicos da Libras.

Outro trabalho em vista é o desenvolvimento de uma aplicação do sistema na forma de um jogo de datilologia em Libras. Esse jogo poderá ter o seguinte formato: após iniciado, o sistema irá exibir em um *display* as palavras que deverão ser datilogradas. O usuário executa os sinais correspondentes às letras do alfabeto que datilogram a palavra exibida e o sistema retorna uma pontuação de acordo com a taxa de acerto ou a velocidade de gesticulação da palavra. Com

isso, acredita-se que o sistema possa contribuir ainda mais com o processo de aprendizagem do alfabeto, uma vez que estimula que o usuário se esforce para adquirir maior pontuação. Da mesma maneira, a competição entre diferentes usuários também pode ser um fator para contribuir com o aprendizado.

## REFERÊNCIAS

ALMEIDA, Josiane J. F. D. **Libras na Formação de Professores: Percepções dos Alunos e da Professora**. 2012.[s.n.], Londrina, 2012.

ALMEIDA, Magno P.; ALMEIDA, Miguel E. História de Libras: Características e sua estrutura. **Philologus**, Rio de Janeiro, v. 18, n. 54, p. 315-327, Setembro 2012.

ATG LTD. **Catálogo MaxiFlex® Ultimate®**. Disponível em: <[http://www.atg-glovesolutions.com/dms/10/Catalogue\\_MaxiFlex\\_Ultimate\\_br\\_20141021.pdf](http://www.atg-glovesolutions.com/dms/10/Catalogue_MaxiFlex_Ultimate_br_20141021.pdf)>. Acesso em: 11 Março 2016.

BALL, Stuart R. **Embedded Microprocessors Systems: Real Word Design**. 3ª. ed. [S.I.]: Newnes, 2002.

BOLTON, Willian. **Instrumentação & Controle**. [S.I.]: Hemus, 2005.

BONACORSO, Nelson G.; NOLL, Valdir. **Automação Eletropneumática**. 10ª. ed. São Paulo: Érica, 1997.

BOYLESTAD, Robert L.; NASHESKY, Louis. **Dispositivos Eletrônicos**. 11. ed. São Paulo: Pearson, v. Único, 2013.

BRASIL. **Lei 10.436, de 25 de abril de 2002**. 2002.[s.n.], Brasília-DF, 2002.

CARVALHO, Dariel. **Software em Língua Portuguesa/Libras com Tecnologia de Realidade Aumentada**. 2011. 144p.Tese(Doutorado em Educação) - Universidade Estadual Paulista, Marília, 2011.

CASSIOLATO, César. Condicionamento de sinais analógicos e sensores. **Saber Eletrônica**, São Paulo, v. 47, n. 460, p. 22-30, Mar/Abr 2012.

CASTRO, Alberto R. D.; CARVALHO, Ilza S. **Comunicação por Língua Brasileira de Sinais**. 3ª. ed. Guará: SENAC, 2009.

DAVIES, John D. **MSP Microcontroller Basics**. 1ª. ed. [S.I.]: Newnes, 2008.

DFROBOT. **DFRobot - Quality Arduino and Robotics Products**. Disponível em: <[http://www.dfrobot.com/wiki/index.php/LCD\\_KeyPad\\_Shield\\_For\\_Arduino\\_SKU:\\_DFR0009](http://www.dfrobot.com/wiki/index.php/LCD_KeyPad_Shield_For_Arduino_SKU:_DFR0009)>. Acesso em: 4 Abril 2016.

DUARTE, Francisco J. A. **Classificação de atividades físicas através do uso do acelerômetro do Smartphone**. Instituto Superior de Engenharia de Lisboa. Lisboa. 2013.

DUBEY, Rahul. **Introduction to Embedded System Design using Field Programable Gate Arrays**. Londres: Springer, 2009.

DUCU, Dragos. **Op Amp Rectifiers, Peak Detectors and Clamps**. 2011. Microchip Technology Inc., Chandler, AZ, 2011.

DUDA, Richard O.; HART, E. P.; STOCK, David, G. **Pattern Classification**. 2<sup>a</sup>. ed. [S.I.]: Wiley-Interscience, 2001.

FACELLI, Katti et al. **Inteligência Artificial: uma abordagem de Aprendizado de Máquina**. Rio de Janeiro: LTC, 2011.

FAHN, Chin-Shyurng; SUN, Herman. Development of a data glove with reducing sensors based on magnetic induction. **IEEE Transactions on Industry Electronics**, v. 52, p. 585-594, Abril 2005.

FAMBRINI, Fabrício. **Library de Manipulação de LCDs**. Disponível em: <<https://groups.google.com/forum/#!topic/hive76-discussion/4PNA7SmirK0>>. Acesso em: 14 Março 2016.

FERREIRA, Lucinda. **Por uma Gramática de Línguas de Sinais**. 2<sup>a</sup>. ed. Rio de Janeiro: Tempo Brasileiro, v. único, 2010.

FERREIRA-BRITO, Lucinda; LANGEVIN, Remi. **Por uma Gramática de Línguas de Sinais**. Rio de Janeiro: Tempo Brasileiro, 1995.

FIGUEIRA, Alexandre D. S. **Material de Apoio para o Aprendizado de LIBRAS**. São Paulo: Phorte, 2011.

FILHO, Ogê M.; NETO, Hugo V. **Processamento Digital de Imagens**. Rio de Janeiro: BrassPort, 1999. 1-5 p.

FREESCALE SEMICONDUCTOR INC. **M68HC05 Family: Understanding Small Microcontrollers**. 1998.[s.n.], 1998.

GESSER, Audrei. **LIBRAS? que língua é essa?** 1<sup>a</sup>. ed. São Paulo: Parábola, v. único, 1971.

GREENLEAF, W.J. Rehabilitation Ergonomics and Disability SOLUTIONS using Virtual Reality Technology. **Proceedings of Interactive Technology and New Paradigm for Healthcare Conference**, San Diego, p. 415-422, 1995.

GROMALA, D.; SHARIR, Y. Dancing with the Virtual Dervish: virtual bodies. **Proceedings of Virtual Reality Software and Technology Conference**, Singapore, p. 321-328, 1994.

HALLIDAY, David; RESNICK, Robert; WALKER, Jearl. **Fundamentos de Física**. 8<sup>a</sup>. ed. Rio de Janeiro: LTC, v. 3, 2009.

HAYKIN, Simon. **Redes Neurais: princípios e prática**. 2<sup>a</sup>. ed. Porto Alegre: Bookman, 2001.

IBGE - INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Censo Demográfico 2010: Características gerais da população, religião e pessoas com deficiência**. São Paulo, p. 12. 2010.

JAIN, Jina; DEODHAR, Leena, PARMAR, P.; MAHESHKAR, Vaishali. Implementation of Fingerspelling Glove. **International Journal of Engineering Trends and Technology**, v. 9, n. 14, p. 732-735, Março 2014.

JAQUES, M.W.S; STRICKLAND, P; OLIVER, T.J. Designe by Manufacturing Simulation: concurrent engineering meets virtual reality. **Proceedings of Joint Brith-Hungarian Mechatronics Conference**, Budapest, Hungary, p. 637-642, 1994.

KHAN, Shujjat et al. Sign Language Analysis and Recognition: A Preliminary Investigation. **24th International Conference Image and Vision Computing**, New Zealand, n. 24, 2009.

LEITE, Tarsísio D. A.; MCCLEARY, Lealand. **Estudos Diarios: fatores complicadores e facilitadores no processo de aprendizagem da Língua Brasileira de Sinais por um adulto ouvinte**. IN. 2001. 242-276p. Arara Azul, São Paulo, 2001.

LUGER, Gorge, F. **Inteligência Artificial**. 6ª. ed. São Paulo: Pearson, 2013.

MALVINO, Albert P.; BATES, David P. **Eletrônica**. 7ª. ed. São Paulo: McGraw-Hill, v. 1, 2007.

MARTINS, Nardênio A. **Sistemas Microcontrolados: Uma abordagem com o microcontrolador PIC 16f84**. São Paulo: Novatech, 2005.

MARWEDEL, Peter. **Embedded System Design**. 2º. ed. Londres: Springer, 2011.

MEDEIROS, Luciano F. D. **Redes Neurais em Delphi**. 1ª. ed. Florianópolis: Visual Books, v. único, 2003.

MOHANDÉS, Mohamed; DERICHE, Mohamed; LIU, Ji. Image-Based and Sensor-Based Approaches to Arabic Sign Language Recognition. **IEEE Transaction on Human-Machine Systems**, p. 2168-2291, Abril 2014.

OLIVEIRA, André S. D. **Sistemas Embarcados: Hardware e Firmware na prática**. 1ª. ed. [S.I.]: Érica, 2006.

OZ, Cemil; LEU, Ming C. American Sign Language word recognition with a sensory glove using artificial neural networks. **Engineering Applications of Artificial Intelligence**, v. 24, p. 1204-1213, 2011.

PALLÀS, Ramon; CASAS, Òscar; BRAGÓS, Ramon. **Sensores y acondicionadores de señal: problemas resueltos**. Barcelona: ES: Marcombo, 2008.

PALLÀS-ARENY, Ramón; WEBSTER, John G. **Sensors and signal conditioning**. 1ª. ed. Nova York: J. Wiley, 1991.

PAZOS, Fernando. **Automação de Sistemas & Robótica**. Rio de Janeiro: Axcel Books, 2002.

PEREIRA, Fábio. **Microcontroladores MSP430**. 1ª. ed. São Paulo: Érica, 2005.

RAMOS, Clélia R. **LIBRAS: A Língua de Sinais dos Surdos Brasileiros**. Disponível em: <<http://www.editora-arara-azul.com.br/pdf/artigo2.pdf>>. Acesso em: 01 Novembro 2014.

REBOLLAR, Hernández J. L.; LINDEMAN, R.W.; KYRIAKOPOULOS, N. A MultiClass Pattern Recognition System for Practical Finger Spelling Translation. **Fourth Int'l Conference on Multimodal Interfaces (ICMI'02)**, Pittsburg, EUA, n. 4, p. 185-190, Outubro 2002.

ROCHA, Fábio S. D. Acelerômetro eletrônico e a placa Arduíno para ensino de Física em tempo real. **Caderno Brasileiro de Ensino de Física**, v. 31, p. 98-123, Abril 2014.

ROSA, Maria J. A. Inclusão: A importância do uso da LIBRAS nas Escolas Públicas Regulares Para Alunos Surdos. **III FÓRUM IDENTIDADES E ALTERIDADES**, Itabaina, n. 3, Novembro 2009.

ROSENBERG, R; SLATER, M. The Chording Glove: a glove-based text input device. **IEEE Trans. System Man Cybern**, p. 186-191, 1999.

RUSSEL, Stuart; NORVIG, Peter. **Inteligência Artificial**. 2ª. ed. [S.I.]: Elsevier, 2004.

SANTANA, Ana P. **Surdez e Linguagem**. São Paulo: Plexus, 2007.

SILVA et al. **Redes Neurais Artificiais para engenharia e ciências aplicadas**. São Paulo: Artliber, 2010.

SILVA, Alexandre F. D.; GONÇALVES, Anselmo F.; MENDES, Paulo M. FBG Sensing Glove for Monitoring Hand Posture. **IEEE Sensors Journal**, v. 11, n. 10, Outubro 2011.

SPECTRA SYMBOL. **Flex Sensor Datasheet**. Disponível em: <<http://media.digikey.com/pdf/Data%20Sheets/Spectra%20Symbol/FS%20Series%20Flex%20Sensor.pdf>>. Acesso em: 25 Novembro 2014.

STURMAN, D. J.; MEDIALAB, Paris; ZELTER, D. A Survey of Gloves-based Input. **Computer Graphics and Applications**, v. 14, p. 30-39, Janeiro 1994.

TANYAWIWAT, Netchanok; THIEMJARUS, Surapa. Design of an Assistive Communication Glove using Combined Sensory Channels. **International Conference on Wearable and Implantable Body Sensor Networks**, n. 9, p. 34-39, 2012.

TEXAS INSTRUMENTS INC. **MSP430F5529 LaunchPad™ Development Kit (MSP-EXP430F5529LP)** (Rev. C). Disponível em: <<http://www.ti.com/lit/ug/slau533c/slau533c.pdf>>. Acesso em: 11 Março 2016.

TUNG, C. P.; KAK, A.C. Automatic Learning of Assembly Tasks using a Data Glove System. **International Conference Intelligence Robots and System**, Pittsburgh, p. 1-8, 1995.

XU, Ruize; ZHOU, Shengli; LI, Wen J. MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognition. **IEEE SENSORS JOURNAL**, v. 12, n. 5, p. 1166-1173, Maio 2012.

YOUNG, Hugh D.; FREEDMAN, Roger A. **Física III: Eletromagnetismo**. 12ª. ed. São Paulo: Pearson, v. III, 2009.

## APÊNDICES

### APÊNDICE A

A seguir é apresentado os códigos complementares às listagem mostradas na Seção 4.5. Na listagem 6 é apresentada os códigos utilizados no seleção dos canais dos geradores e sensores.

```
// variável que define o numero de iterações da varredura
#define iteracoesVarredura 13

// define a ordem de ativação dos geradores
const int vtSelecaoGerador[] =
    {G1, G1, G1, G1, G1, G2, G3, G3, G3, G3, G4, G5, G6 };
// define a ordem de seleção dos senores
const int vtSelecaoSensor[] =
    {S1, S2, S3, S4, S5, S5, S1, S2, S3, S6, S6, S7, S8 };

// define os valores que configuram as frequências de ressonância dos pares de bobinas
int vetpwm[] =
    {123, 120, 119, 118, 121, 120, 128, 128, 127, 125, 128, 125, 128 };

// função de seleção da porta que seleciona o canal do sensor x
void _SelecionaSensor(unsigned int x) {
    if (x == 0 || x == 6)
        PORTCHAN2 &= S1;
    else
        PORTCHAN = vtSelecaoSensor[x];
}

// função de seleção da porta que seleciona o canal do gerador x
void _SelecionaGerador(unsigned int x) {
    PORTMUX &= ~MUX_BASEPIN;
    PORTMUX |= vtSelecaoGerador[x];
}

// função de seleção da frequência de ressonância
void _SelecionaFrequencia(unsigned int x){
    TAICCR0 = vetpwm[x]; // configura frequencia
    TAICCR1 = TAICCR0 >> 1; // ajusta duty cycle 50%
}

```

Listagem 6 - Códigos das funções de seleção de canais.

Na listagem 7 é apresentado os códigos que implementam a função de normalização. Essa função é baseada na função “mapminmax” do MATLAB.

```
// define os parâmetros da função de Normalizacao
#define NORMALIZA 0
#define DESNORMALIZA 1

// função de normalização
float normaliza(float x, unsigned int i, unsigned int op) {

    float ret;
    if (op == NORMALIZA) {
// remapeia dados para valores entre 1 e menos 1

        if (xmi n[i] != xmax[i])
            ret = ((2) * (x - xmi n[i]) / (xmax[i] - xmi n[i])) - 1;
        else
            ret = x;

    } else if (op == DESNORMALIZA) {
// remapeia dados para valores entre zero e um

        if (xmi n[i] != xmax[i])
            ret = ((1 - 0) * (x + 1) / (2)) - 0;
        else
            ret = x;
    } else
        ret = x;
    return ret;
}
```

Listagem 7 - Código da função de normalização.