



**Afonso de Azevedo e Silva Nunes Pedroso**

Licenciado em Ciências de Engenharia Eletrotécnica e de Computadores

## **Tech4Good – Interpretador de LGP**

Dissertação para obtenção do Grau de Mestre em  
Engenharia Eletrotécnica e de Computadores

Orientador: Professora Doutora Anikó Katalin Horváth da Costa

Co-orientadores: Doutor Pedro Marques Couto

Júri:

Presidente: Professor Doutor André Teixeira Bento Damas Mora

Vogais: Professora Doutora Anikó Katalin Horváth da Costa

Professor Doutor Tiago Oliveira Machado de Figueiredo Cardoso

**Novembro 2019**



### **Tech4Good – Interpretador LGP**

Copyright © Afonso de Azevedo e Silva Nunes Pedroso, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja crédito ao autor e editor.



*Dedicado à minha família*





## Resumo

Nos dias que correm, a necessidade de transmitir informação, entre pessoas, ou entre dispositivos é algo indispensável. A necessidade de uma comunicação fácil é cada vez mais importante, quer do ponto de vista da sociedade quer do ponto de vista comercial. No entanto, quando falamos na perspetiva de alguém que possui dificuldades auditivas, ou até mesmo se for surda, a comunicação é dificultada de uma forma exponencial.

Apesar de este problema afetar uma minoria da população, não deixa de ser um problema eminente e que, com certeza, causará desconforto tanto à pessoa em questão, que sente frustração por não conseguir expressar-se e fazer-se entender, situação que lhe provoca uma baixa de autoestima, como aos seus familiares e amigos, que se sentem impotentes por não conseguirem ajudar a pessoa no seu problema, nem compreender as suas necessidades, ideias e pensamentos.

Esta tese de mestrado surge precisamente na ótica de uma possível solução para este problema. Propõe-se o desenvolvimento de uma luva, com vários sensores integrados que, posteriormente, irão medir a posição relativa da mão no espaço. A partir do movimento da mão, transformamos os dados em letras do alfabeto de língua gestual portuguesa.

Após o reconhecimento dos caracteres, a ideia que preconizamos é transformar estes sinais dos movimentos da mão, para voz. Este processo acontece através do serviço Amazon *Polly*, ou Google Cloud *Text-To-Speech*, de modo a que tais inputs possam ser audíveis e compreendidos por toda a gente. Surdos e não surdos. Uma tecnologia inclusiva, em que todos, independentemente da sua deficiência ficariam em igualdade de circunstâncias.

**Palavra chave:** Portadores de Deficiência Auditiva, Comunicação, Luva, Sensores, Text-To-Speech



# Abstract

Nowadays, the need to transmit information, be it between people or between devices is getting more and more important. The need of communication and its inherent ease is something important in a social perspective. However, when we are talking about someone's perspective that either is hard of hearing or totally deaf, this communication is hardened in an exponential form.

Although this problem affects a minority of people, it doesn't stop being an eminent problem that will certainly lead to discomfort, either to the person that has the hearing impairment, that may feel frustration due to not being able to express himself, either to their family and friends, that are there and may feel unable to help them with their problem.

This master's thesis arises precisely in the perspective of a possible solution to this problem. The project consists in the creation of a glove, that has various built-in sensors that will measure hand position related to the space, so that when the hand moves the data can be converted to LGP (*Portuguese Sign Language*).

When the recognition of the letters is done the main purpose is to convert this inputs to voice, with the aid of a text-to-speech engine like Amazon *Polly* or Google cloud *text-to-speech*, so that this inputs can be heard and understood between all people.

**Key words:** Communication, Deafness, Glove, Sensors, *text-to-speech*

---



## ÍNDICE DE FIGURAS

1. INTRODUÇÃO .....	1
1.1 ENQUADRAMENTO .....	1
1.2 MOTIVAÇÃO .....	2
1.3 OBJETIVOS .....	3
1.4 CONTRIBUIÇÕES .....	3
1.5 ESTRUTURA DO DOCUMENTO .....	4
2. ESTADO DE ARTE.....	7
2.1 LÍNGUA GESTUAL PORTUGUESA .....	7
2.2 SOLUÇÕES RELEVANTES .....	11
2.3 INTERPRETADOR LGP .....	17
3. ARQUITETURA DO INTERPRETADOR LGP .....	26
3.1 ARQUITETURA DO SISTEMA .....	26
4. IMPLEMENTAÇÃO DO PROTÓTIPO.....	33
4.1 MONTAGEM DE ARQUITETURA.....	33
4.2 ESCOLHA DE RESISTÊNCIAS .....	34
4.3 AQUISIÇÃO DE DADOS .....	36
4.4 TRATAMENTO DE DADOS .....	43
4.5 DIAGRAMA COMPORTAMENTAL FUNCIONAMENTO DO PROTÓTIPO.....	50
5. TESTES E RESULTADOS.....	53
5.1 TESTES DE AQUISIÇÃO DE DADOS.....	53
5.2 TRATAMENTO DE DADOS .....	54
6. CONCLUSÕES E TRABALHOS FUTUROS.....	57
6.1 CONCLUSÕES .....	57
6.2 TRABALHOS FUTUROS.....	58





# Índice de figuras

FIGURA 2.1 - CARACTERES LÍNGUA GESTUAL PORTUGUESA.....	8
FIGURA 2.2 - CONFIGURAÇÕES DA MÃO EM LGP.....	9
FIGURA 2.3 - CONJUNTO DE POSIÇÕES DA MÃO PARA A PALAVRA CAF.....	9
FIGURA 2.4 - PALAVRA "GALO" VS "GALINHA" .....	10
FIGURA 2.5 - UTILIZAÇÃO DO PROTÓTIPO.....	13
FIGURA 2.6 - FUNCIONAMENTO DA UNI.....	14
FIGURA 2.7 - IMAGEM PROCESSADA RELATIVA AO GESTO.....	15
FIGURA 2.8 - FLUXOGRAMA DO SISTEMA DE RECONHECIMENTO DE LÍNGUA GESTURAL KURDA.....	16
FIGURA 2.9 - OPTICAL <i>FLEX SENSOR</i> .....	18
FIGURA 2.10 - <i>FLEX SENSOR</i> BASEADO EM TINTA CONDUTORA.....	19
FIGURA 2.11 - <i>FLEX SENSOR</i> CAPACITIVO.....	19
FIGURA 2.12 - SENSORES DE ELETROMIOGRAFIA.....	20
FIGURA 2.13 - ARDUINO UNO REV3 .....	21
FIGURA 2.14 - RASPBERRYPI 3B+.....	23
FIGURA 2.15 - ESQUEMA DA COMUNICAÇÃO.....	24
FIGURA 3.1 - SISTEMA DE AQUISIÇÃO DE DADOS.....	27
FIGURA 4.1 - MONTAGEM COMPLETA.....	34
FIGURA 4.2 - TABELA DE VALORES DE RESISTÊNCIA PARA O DEDO MINDINHO.....	35
FIGURA 4.3 - I2C ADDRESS MPU6050.....	37
FIGURA 4.4 - REGISTO 0x6B.....	38
FIGURA 4.5 - FORÇAS APLICADAS QUANDO O SENSOR ESTÁ NUMA POSIÇÃO PLANA.....	38
FIGURA 4.6 - MPU INCLINADO SOBRE O EIXO DOS Y.....	39
FIGURA 4.7 - MPU INCLINADO SOBRE O EIXO DOS X.....	39
FIGURA 4.8 - VALORES RECEBIDOS EXECUTANDO A FORMA DA LETRA B.....	42
FIGURA 4.9 - VALORES DO GIROSCÓPIO RELATIVOS À LETRA "B" .....	42
FIGURA 4.10 - DIAGRAMA COMPORTAMENTAL RELATIVO À AQUISIÇÃO DE DADOS.....	43
FIGURA 4.11 - INTERAÇÃO ARDUINO RASPBERRYPI.....	44
FIGURA 4.12 - VALORES DOS FLEX SENSORES NO RASPBERRYPI ENVIADOS PELO ARDUINO.....	45
FIGURA 4.13 - BASE DE DADOS.....	47
FIGURA 4.14 - OUTPUT COM DISTÂNCIAS ABSOLUTAS RELATIVO À INFORMAÇÃO ENVIADA .....	48
FIGURA 4.15 - EXEMPLO DE LISTA DE FREQUÊNCIA.....	50
FIGURA 4.16 - FUNCIONAMENTO DO PROTÓTIPO.....	51
FIGURA 5.1 - AQUISIÇÃO DA LETRA "B" E "X".....	54



# Lista de Tabelas

TABELA 1 - ESPECIFICAÇÕES DO RASPBERRYPI 3B+.....	22
TABELA 2 - TABELA DE SENSIBILIDADES.....	23
TABELA 3 - MAPEAMENTO DE VALORES CONSOANTE O ÂNGULO DOS <i>FLEX SENSORS</i> .....	40
TABELA 4 - TESTE AQUISIÇÃO DE LETRAS.....	41

## **Acrónimos e Siglas**

ASL – American Sign Language

CPS – Cyber Physical Systems

DMP – Digital Motion Processor

EMG – Electromyography Sensor

ICSP – In circuit Serial Programming IDE – Integrated Development Environment

IOT – Internet of things

I2C – Inter-integrated Circuit

LGP – Língua Gestual Portuguesa

MCU – microcontroller

PWM – Pulse Width Modulation

SCL – Serial Clock

SPI – Serial Peripheral Interface

SDA – Serial Data

UART – Universal Asynchronous Receiver Transmitter

# Introdução

## 1

Esta dissertação enquadra-se no conceito “*Tech4Good*” [3] que tem como objetivo promover a inclusão. Num mundo ideal todos teríamos as mesmas oportunidades, do ponto de vista social, económico, cultural, etc. Neste projeto, o foco é ajudar aqueles que, à partida, independentemente do contexto socioeconómico, têm um *handicap*. Não ouvem. Ou não ouvem com a clareza dos ouvintes. Se vivêssemos num mundo inclusivo, no qual, para lá das características de cada um, as oportunidades fossem iguais para todos, não teríamos este problema de comunicação e de compreensão. No entanto, sabemos que na realidade, existem grupos que infelizmente não conseguem aproveitar todas as oportunidades e benefícios que a tecnologia tem para oferecer, seja por desconhecimento, seja por razões outras que, no âmbito deste trabalho, não importa explicitar. Para se poder lidar com esta situação, há várias iniciativas dirigidas quer a pessoas desfavorecidas e/ou portadoras de deficiência auditiva que, com a ajuda da tecnologia que tem vindo a ser desenvolvida, terão a possibilidade de dispor de um auxiliar que poderá atuar como um verdadeiro agente de transformação das suas vidas.

## 1.1 Enquadramento

Para entender melhor este conceito de “*Tech4Good*”, temos um exemplo da app Drishti [4], que foi desenvolvida por uma grande empresa de consultoria no seu laboratório de I&D da Índia. Esta app permite a portadores de deficiência

visual, com o apoio de uma pequena câmara e com a ajuda de interpretadores de imagem suportados por inteligência artificial, entenderem o que se passa à sua volta. Como por exemplo:

- identificar pessoas,
- poder ler pequenos textos,
- reconhecer diferentes notas e moedas.

Estas atividades que muitos de nós fazemos sem pensar e sem acarretar qualquer esforço, não são evidentes para aqueles que possuem este tipo de deficiência.

Continuando com a ideia do “*Tech4Good*” em mente, a tecnologia ao serviço do bem, propõe-se fazer uma adaptação da ideia já antes referida, através da aplicação *Drishti*, neste caso com o foco no suporte de portadores de deficiência auditiva que utilizem a Língua Gestual Portuguesa (LGP) como forma de comunicação.

A Língua Gestual Portuguesa é o idioma através do qual a maior parte da comunidade portuguesa que é surda, comunica entre si. É processada através de gestos que estão caracterizados e pela sua captação visual. É utilizada pela comunidade surda que, em Portugal, rondará os trinta mil portadores de deficiência auditiva [5].

## **1.2 Motivação**

A tecnologia, de um modo geral, tem evoluído bastante. O seu desenvolvimento aliado à capacidade de atribuir às tecnologias de inteligência, com os sistemas capazes de agir conforme as necessidades do utilizador, têm proporcionado soluções que se revelaram de enorme utilidade para esta população. É aqui que entram os CPS (*Cyber Physical Systems*) [6], sistemas que são utilizados quando se pretende que os sistemas físicos ou complexos interajam, ou comuniquem com o mundo digital, de modo a permitir que o seu desempenho seja otimizado, assim como a sua eficiência. Os CPS desempenham

um papel bastante importante no processo industrial e no controlo de produção, as chamadas “*smart factories*” [7], em particular no caso das IOT (*Internet of things*) [8].

A manifestação da tecnologia IOT promoveu a melhoria dos diversos sistemas tanto no que toca ao desempenho, assim como na monitorização. Agora estamos perante um sistema de partilha de informação, que vai ajudar na comunicação entre os dispositivos.

É nesta vertente que surge a motivação da realização deste trabalho. Não descurando o foco principal que é o “*Tech4Good*”, já referido. A elaboração deste projeto passa também pelo desenvolvimento de um sistema que seja capaz de agregar, o sistema em questão e de lidar com as várias tecnologias inerentes ao sucesso do produto.

O projeto em questão está a ser desenvolvido em parceria com a empresa Accenture, que propôs este tema.

### **1.3 Objetivos**

Esta dissertação vem no sentido de ajudar as pessoas que sofrem de problemas auditivos e/ou surdez a comunicarem com a sociedade, sendo que a pergunta que surge é “Como se poderá auxiliar indivíduos portadores de deficiência auditiva na sua comunicação com a sociedade?”.

Existem dois tipos de tecnologias que se destacam para auxiliar pessoas com estes problemas. As primeiras tecnologias são as que ajudam o indivíduo a melhorar o seu estado e/ou curar o seu problema. O segundo tipo são as tecnologias que apesar de não terem o intuito de melhorar a sua condição auditiva, são uma grande ajuda para promover a comunicação com a sociedade.

### **1.4 Contribuições**

As contribuições desta dissertação focam-se na elaboração de uma luva, que irá ser usada por um indivíduo, onde este irá efetuar o gesto necessário de modo

reproduzir letras do alfabeto LGP e agregá-las, de modo a construir palavras, com o intuito final de reproduzir tais palavras em voz.

A luva terá na sua construção sensores para cada dedo, que deverão ter as seguintes capacidades:

- Configuração das mãos;
- Local da articulação;
- Movimento das mãos;
- Orientação das mãos;

Ainda se considerou a **possibilidade** para um quinto parâmetro da GLP (Componente não manual, expressão e movimento corporal), onde será avaliada a contingência de integração de um dispositivo com câmara para a captação da cara do gestuante, este ponto acabou por não ser implementado nesta dissertação.

Com base nos dados adquiridos pelos sensores, o sistema é capaz de efetuar a sua tradução para LGP e posteriormente utilizar um serviço em Cloud (ex: Amazon *Polly* [1], Google Cloud *text-to-speech* [2], ou semelhante) para vocalizar o caractere pretendido.

## 1.5 Estrutura do documento

O presente documento encontra-se dividido em 6 principais capítulos.

No segundo capítulo, é feito o levantamento bibliográfico do tema abordado. Neste capítulo serão apresentadas soluções semelhantes à presente, bem como uma apresentação relativa a Língua Gestual Portuguesa e problemas relacionados com a deficiência auditiva.

O terceiro capítulo tem como principal objetivo apresentar a arquitetura do protótipo, bem como todos os dispositivos em detalhe.

No quarto capítulo é apresentada a implementação do sistema onde é possível compreender todo o funcionamento do protótipo. Pode-se também observar a montagem de toda a arquitetura.

No quinto capítulo são apresentados testes e resultados experimentais realizados relativos ao protótipo.

O sexto e último capítulo destina-se à apresentação das conclusões relativas à realização deste projeto, bem como todo o trabalho futuro e limitações do sistema.



# Estado de Arte

## 2

Serve este presente capítulo para dar a entender alguns conceitos derivados da língua gestual e os problemas que as pessoas com deficiência auditiva, poderão enfrentar ao longo das suas vidas. Será também dada ênfase no que toca a apresentação de tecnologias semelhantes à que irá ser apresentada desta dissertação. Poderemos também encontrar uma explicação um pouco mais específica da tecnologia apresentada neste projeto.

### 2.1 Língua Gestual Portuguesa

A Língua Gestual Portuguesa é encarada, ou pelo menos deve ser encarada como elemento para-linguístico, ou seja, elemento para lá da comunicação, sendo utilizada como instrumento de apoio à comunicação oral, tendo assim uma elevada contribuição para um certo grau de expressividade do emissor. Este nível de expressividade não deve ser tomado como absoluto, o referido valor deve ser relativizado devido ao facto de este depender do ponto de vista do recetor e da situação comunicacional[9]. As línguas gestuais devem ser encaradas como línguas humanas, na medida que terão de obedecer a parâmetros linguísticos universais, assim como a arbitrariedade, a convencionalidade, a recursividade e a criatividade.

Pelo que é observado, as línguas gestuais assumem características intrínsecas a cada uma que as distingue, havendo assim como no caso das línguas orais um certo grau de pluralidade de idiomas. Para provar esta constatação temos o exemplo dos gestos, que obviamente serão distintos tendo em

consideração a diferença entre cada idioma, é ainda de salientar que as línguas gestuais são a imagem e a capacidade criadora das línguas humanas, visto que novos vocábulos vão aparecendo à medida que a necessidade de definir certos conceitos e novas realidades se torna eminente.

### 2.1.1 Configuração do Gesto em Língua Gestual Portuguesa

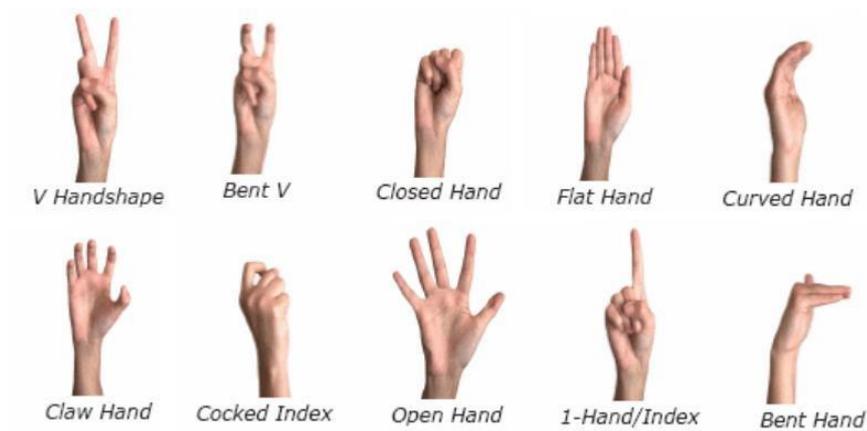
Em relação à evolução da Língua Gestual, William Stokoe, a meados do século XX impulsionou o desenvolvimento das línguas gestuais, passando apenas de uma representação mímica para um sistema linguístico. Segundo este panorama linguístico, a Língua Gestual é formada por cinco unidades de configuração possíveis para o gesto, sendo estes: a configuração da mão, o local da articulação, a orientação da mão, o movimento da mão e a expressão facial [10].

**Configuração da mão:** Na Figura 2.1, podemos observar as possíveis configurações da mão na língua gestual que está relacionada com as diferentes posições que os dedos podem adotar, que servirão para a representação de caracteres.



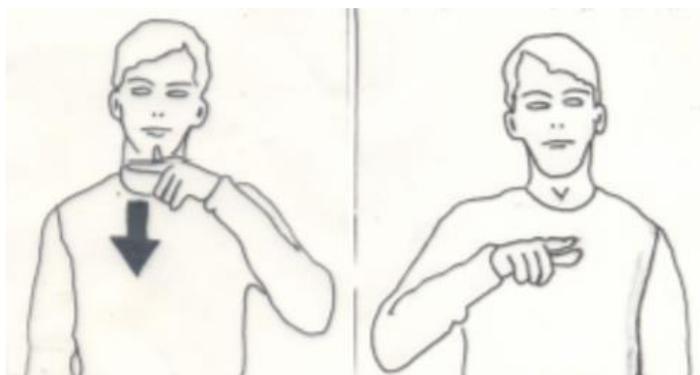
Figura 2.1 - Caracteres Língua Gestual Portuguesa[11]

Na Figura 2.2 são apresentadas algumas configurações possíveis da orientação da mão, no entanto não se encontram representadas na totalidade:



**Figura 2.2 - Configurações da mão em LGP [12]**

**Configuração da mão:** A orientação da mão em língua gestual refere-se, como se consegue entender pela expressão, à orientação da palma da mão. Em alguns casos, com apenas uma ligeira mudança na orientação, ficamos com palavras distintas, como podemos ver no exemplo da Figura 2.3:



**Figura 2.3 - Conjunto de posições da mão para a palavra Café (bebida) VS Café (local)**

[13]

**Local da articulação:** O local da articulação na língua gestual indica-nos o local onde a articulação se irá situar no final de um certo tipo de gesto. Temos o exemplo seguinte, em que conseguimos ver que as palavras “Galo” e “Galinha”, possuem a mesma configuração, o mesmo movimento, sendo apenas distintas no que toca ao local da articulação, como podemos verificar na Figura 2.4.

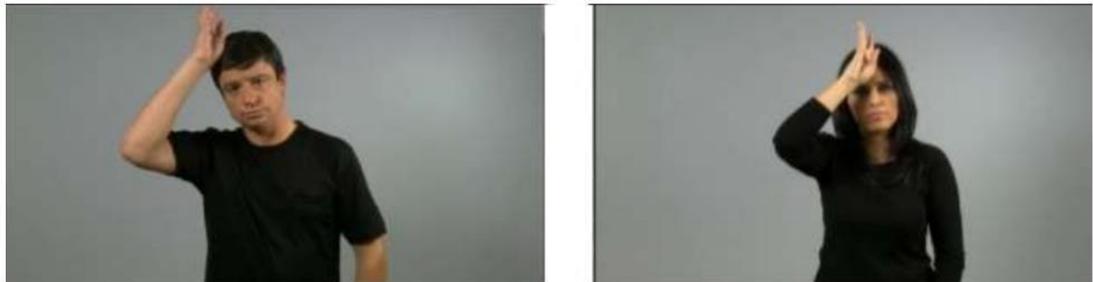


Figura 2.4 - Palavra "Galo" vs "Galinha"[13]

### 2.1.2 Problemas relacionados com a deficiência auditiva

A deficiência auditiva é uma condição heterogénea com efeitos massivos a nível social, emocional e mesmo a nível do desenvolvimento do sistema cognitivo [14]. A perda auditiva afeta cerca de 15-26%, onde os mais prejudicados são os países com menos possibilidades financeiras. A porção de pessoas que sofrem deste problema tem vindo a aumentar, embora seja uma minoria a densidade de população que usa a língua gestual. Podemos identificar vários problemas, como podemos verificar de seguida:

- No caso dos EUA cerca de 27% dos deficientes auditivos compreendidos em idades entre os 6 e 19 anos de idade possuem outras debilitações graves, assim como o retardamento do desenvolvimento mental, o desenvolvimento físico, dificuldades de aprendizagem específicas, deficiência visual e

autismo [14]. Crianças com deficiência auditiva agravada aprendem o vocabulário a metade da velocidade a que uma criança que não possua tal deficiência aprenderia [15].

- A abstinência relativa ao acesso de estimulação auditiva e o posterior atraso da assimilação de uma certa língua afeta o processo no domínio neuro cognitivo, assim como a visão, a memória e a atenção da pessoa [16].
- Um estudo baseado numa população generalizada refere que adultos com perda auditiva são cerca de três vezes mais vulneráveis em relação aos restantes indivíduos a possuírem sintomas de psicose [14].
- Os indivíduos que sofrem de deficiência auditiva são cerca de duas vezes mais vulneráveis a experienciar traumas em comparação com a generalidade da população, assim como são mais suscetíveis a serem vítimas de maus tratos. Além disso, em comparação com os outros indivíduos, os deficientes auditivos estão mais vulneráveis a desenvolver problemas de stress pós-traumático [17].

## **2.2 Soluções relevantes**

É, por isso, importante que o acesso à aprendizagem de língua gestual seja fortemente globalizado aliado à provisão de serviço especialista de profissionais com formação específica para a comunicação com pessoas portadoras de deficiência auditiva. Este acesso especializado revelou-se bastante mais eficiente, quando comparado com um serviço mais “regular” em que havia apenas uma aula de apoio para vários estudantes com este problema, ou seja, a atenção especializada a cada pessoa com este tipo de deficiência revela-se bastante importante. Outro aspeto também relevante é a interação entre indivíduos com este tipo de debilitações [20].

Apesar de estudos demonstrarem que um maior desenvolvimento é notado quando o portador de deficiência tem um acompanhamento especializado, em muitos dos casos a criança gostaria de se sentir integrada na sociedade, é neste paradigma que seria bastante importante a implementação de algum tipo de tecnologia que servisse de auxílio a pessoas com este tipo de problemas [21].

### **2.2.1 Protótipo Texas University**

Alguns investigadores na Texas A&M University, desenvolveram um dispositivo que se pode usar no braço, que poderia auxiliar na comunicação “traduzindo” a língua gestual para a americano, neste caso, sentindo os movimentos da pessoa em questão. Este produto é coordenado pelo Dr. Roozbeh Jafari, na Texas A&M University.

Este dispositivo ainda é um protótipo, mas já consegue reconhecer cerca de 40 gestos em ASL (*American Sign Language*) com elevada precisão. Funciona com a utilização de dois sensores um primeiro sensor de movimento utilizando um acelerómetro e um giroscópio para medir a velocidade e ângulo da mão. Obtendo então estes parâmetros, o dispositivo consegue então começar a decifrar as palavras que o utilizador quiser transmitir, enviando então as palavras correspondentes para o smartphone. À parte disto existe ainda um sensor de eletromiografia, que mede os potenciais elétricos do movimento muscular [22].

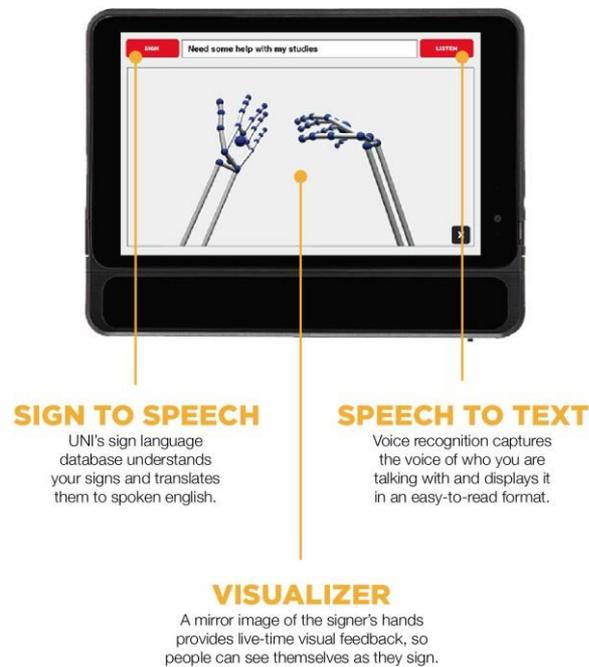


**Figura 2.5 - utilização do protótipo [22]**

### **2.2.2 Uni (MotionSavvy Prototype)**

A empresa norte americana MotionSavvy, criou um produto, denominado por Uni, em formato de tablet que traduz a língua gestual para áudio e texto escrito.

O Uni utiliza uma tecnologia que reconhece os gestos em movimento, permitindo assim aos utilizadores verem tais movimentos reproduzidos no ecrã, movimentos estes captados através de uma câmara, podendo assim limitar o erro e a perda de informação no decorrer de uma conversa. Este tablet vem com um dicionário incluído de ASL, no entanto é dado ao utilizador a hipótese de acrescentar mais sinais. O Uni funciona então através de duas câmaras e um microfone projetando imagens dos gestos da pessoa surda num espaço 3D. Este projeto está ainda em fase de protótipo e compreende cerca de 300 palavras [23].



**Figura 2.6 - Funcionamento da Uni [23]**

### 2.2.3 Kurdish Sign Language Recognition System

Este sistema proposto é baseado em língua gestual curda (KuSL), que todavia, não havia sido introduzido antes. O presente sistema funciona a tempo-real e toma as decisões necessárias assim que gesticulado um movimento. Neste projeto, 3 algoritmos de detecção de KuSL foram implementados e testados, 2 deles foram implementados por outros investigadores fora do projeto, enquanto que um terceiro foi introduzido neste projeto pela primeira vez. Este método chama-se *Grid-based gesture descriptor*, que atingiu uma eficácia de detecção de gestos de cerca de 67%, superior aos outros dois processos visto que esses alcançaram uma eficácia à volta dos 42% [24].

Quanto ao funcionamento do sistema pode-se generalizar o seu funcionamento:

1. 1º Passo: terá que se detetar a forma do gesto que a pessoa quer transmitir, recebido através de uma imagem relativa ao vídeo, posteriormente aplicando uma certa quantidade de técnicas de pré-processamento, assim como redução de ruído, métodos de segmentação de imagem, assim como o *image thresholding* [25] e *background extraction* [26].



**Figura 2.7 - Imagem processada relativa ao gesto [24]**

2. 2º Passo: Encontrar a região que interessa segmentar como podemos observar na Figura 2.7, tendo acesso a algumas ferramentas de extração de algoritmos. Nesta fase o sistema irá reconhecer e distinguir cada sinal da mão consoante o seu significado.
3. 3º Passo: Irão se comparar as informações retidas da imagem de *input* com os gestos que estão guardados numa base de dados e verificar as correspondências. Caso as condições sejam cumpridas, o sistema irá imprimir o devido caracter do alfabeto curdo.

Na Figura 2.8 pode se observar um fluxograma que irá facilitar o entendimento de todo este processo, onde será possível verificar todas as etapas desde a leitura do gesto realizada pela câmara, passando por todas as técnicas de pós processamento de imagem até ao reconhecimento da letra/palavra correspondente e devida impressão.

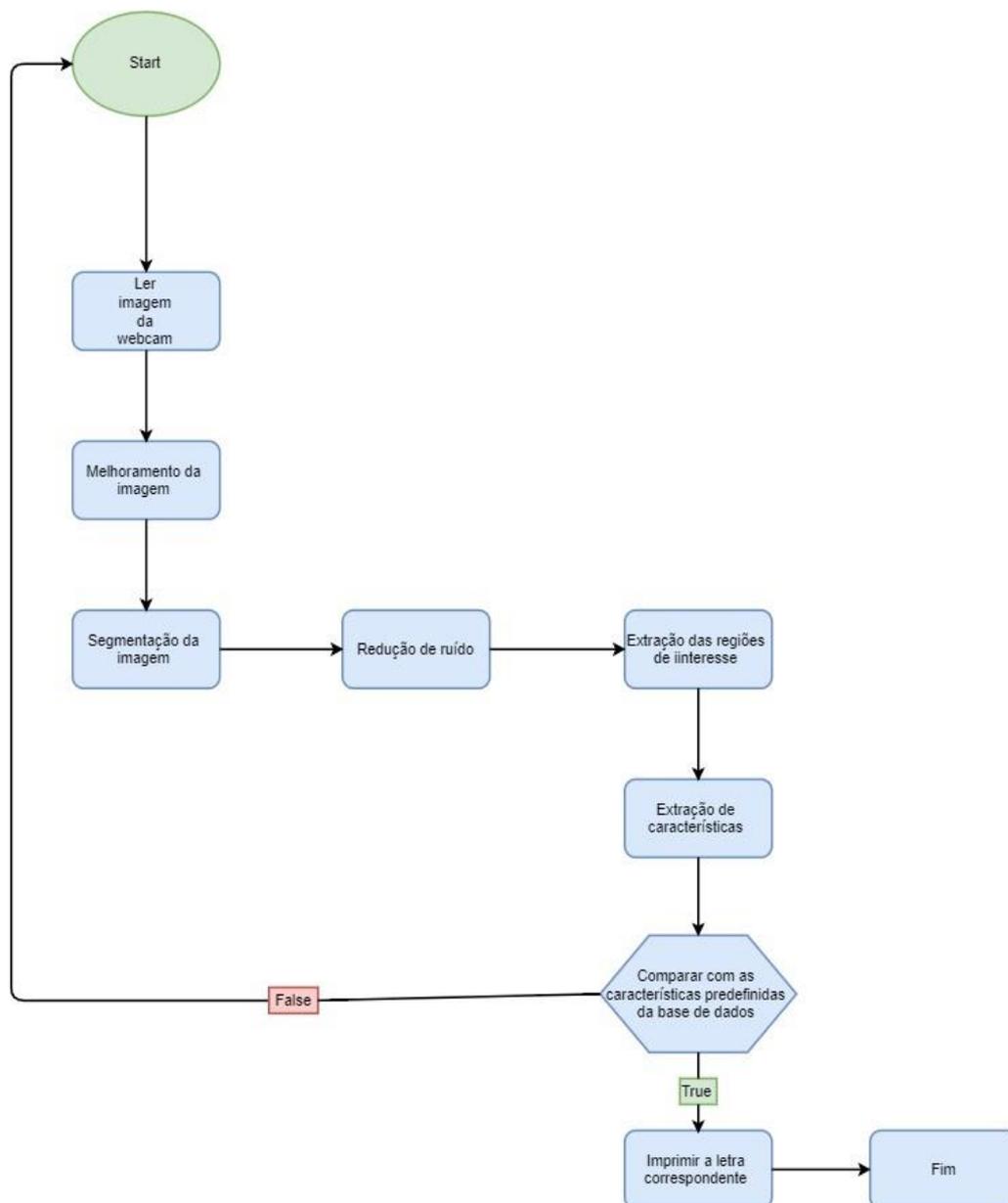


Figura 2.8 - Fluxograma do sistema de reconhecimento de língua gestural Kurda [24]

## 2.3 Interpretador LGP

O interpretador LGP é o tema desta dissertação e como já foi referido é um sistema que irá traduzir Língua Gestual Portuguesa para português, tudo possível através do movimento dos dedos, visto que os sensores estarão integrados na luva.

Para o funcionamento da luva teremos então de estabelecer a aquisição de dados através do controlador, serão necessários um certo tipo de sensores, para a obtenção de informação relativa ao posicionamento da mão e um software programável para a manipulação de dados.

### 2.3.1 Sensores

Os sensores utilizados foram os *Flex Sensors*. Um *Flex Sensor*, também conhecido por *Bend Sensor*, medem a quantidade de deflexão causada pela curvatura do sensor [35].

Os *Flex Sensors* têm múltiplas aplicações, assim como a grande maioria dos sensores. Ainda que estes sejam muito usadas como goniómetro em aplicações de reabilitação, as aplicações transcendem em grande quantia em relação a este paradigma, temos o uso destes sensores em áreas como a geologia e instrumentos musicais. Em cada aplicação o sensor identifica a relação de arqueamento, em relação a uma resistência variável, cujo valor pode ser guardado de forma digital e a informação usada de maneiras distintas dependendo da aplicação em questão. No caso desta aplicação, o facto de se poder obter o arqueamento é crucial para poder obter informação da posição de cada dedo. No caso da geologia, este sensor foi utilizado para identificar zonas onde havia deslizamentos de terra, ou seja, zonas que terrestres que estariam desmoronadas. No caso da música estes sensores são usados na criação de um instrumento de cordas, que pode ser tocado através da deformação e arqueamento das mesmas,

que posteriormente fará a variação de uma resistência e então proporcionar efeitos bastante interessantes.

Existem três tipos de *Flex Sensors* que são os seguintes:

- *Flex Sensors* óticos – Podemos observar a configuração do *Flex Sensor* ótico na Figura 2.9:

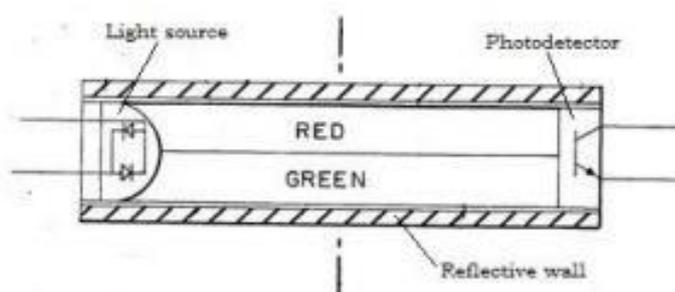
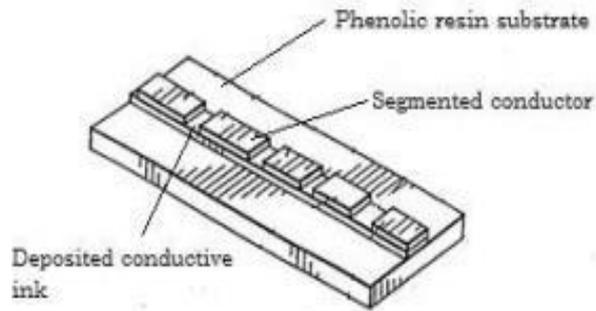


Figura 2.9 - Optical *Flex Sensor* [35]

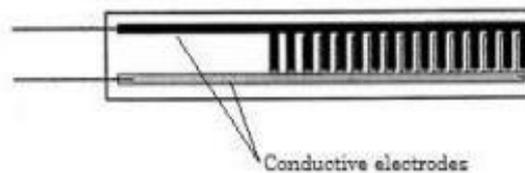
Na Figura 2.9 podemos observar o *Flex Sensors* ótico, que consiste num tubo flexível onde existem duas extremidades, um interior que possui uma parede refletiva, numa das extremidades do tubo flexível reside uma fonte de luz, enquanto que na outra extremidade poderemos encontrar um detetor fotossensível, de maneira a detetar a combinação de raios diretos e refletidos, quando se dá o arqueamento do sensor.

- *Flex Sensor* baseado em tinta condutora - consiste num substrato de resina fenólica com tinta condutora depositada e um condutor segmentado é colocado no topo para formar um potenciômetro flexível no qual a resistência irá variar consoante a flexão. Podemos observar a sua descrição na:



**Figura 2.10 - Flex Sensor baseado em tinta condutora [35]**

- *Flex Sensor* Capacitivo – Esta alternativa foi criada por entidades distintas que as mencionadas acima, neste caso existem duas camadas condutoras separadas por um material dielétrico entre as camadas, reduzindo assim a resistência entre camadas Figura 2.11.



**Figura 2.11 - Flex Sensor capacitivo [35]**

Foi também considerada a utilização de sensores EMG (*Electromyography Sensor*), este tipo de sensores é utilizado para detetar sinais derivados da ativação muscular neste caso do antebraço.

Estes sensores são feitos de um material pouco extensível e agem como atuadores que conforme o movimento da mão geram um sinal elétrico, devido

das contrações musculares, capaz de ser monitorizado, como por exemplo o abrir e fechar da mão [36].

Esta técnica seria bastante interessante de implementar, pois após o reconhecimento do alfabeto LGP, seria uma boa ideia apostar no reconhecimento de expressões o que leva a que o sistema tenha que interpretar também os movimentos do antebraço, onde apenas os *Flex Sensors*, não seriam suficientes.



**Figura 2.12 - Sensores de Eletromiografia [33]**

Na Figura 2.12 podemos observar os elétrodos, estes são dispositivos que permitem a entrada e saída de corrente elétrica e servem para otimizar a condução do sinal, entre a pele da pessoa e o sistema de aquisição de dados e por sua vez identificar o registo da atividade elétrica exercida pelo músculo [34].

Entre a pele da pessoa e o elétrodo usualmente usa-se um gel. Este gel revela-se um bom elemento para a condução do sinal e permite melhorar tanto a aquisição, como também reduz o ruído adicionado ao sinal [31].

### **2.3.2 Controlador**

A principal funcionalidade do controlador será a aquisição dos valores dos sensores de maneira a que estes possam ser manipulados num software programável.

Através do processamento de informação adquirida pelos sensores, é possibilitado o envio dos mesmos para o RaspberryPi, que impulsionará o funcionamento deste dispositivo.

O controlador proposto para ser usado neste projeto é o Arduíno Uno devido ao seu baixo custo, capacidade de processamento aceitável e elevado número de módulos existentes.

### 2.3.2.1 Arduíno Uno



Figura 2.13 - Arduíno Uno Rev3 [27]

Arduíno é uma plataforma Open-Source usada para o desenvolvimento de projetos eletrônicos. O Arduíno alberga uma placa física programável, também conhecida como MCU (*MicroController unit*) e um *software* IDE (*Integrated Development Environment*) que corre no computador, que por sua vez é usado para escrever e enviar código do computador para a placa física [27].

O microcontrolador ATmega328P, possui 14 pinos digitais onde 6 podem ser utilizados como PWM (*Pulse Width Modulation*) *outputs*, 6 inputs analógicos, 1 oscilador de 16 Mhz que está ligado aos pinos do *clock*, que serve de referencia para todos os programas executados no MCU, 1 conexão USB com um circuito *serial-to-USB*, 1 led de *Debug*, uma saída para alimentação e existe ainda um botão de *reset* que permite o recomeço da execução de um programa [28].

Existe também um ICSP (*In circuit Serial Programming*) header, referente a capacidade de programar diretamente os microcontroladores da placa usando o protocolo serial SPI (*Serial Peripheral Interface*), ou seja, na realidade é o protocolo que irá permitir que sejam descarregados programas para o MCU. É ainda possível o uso de canais de comunicação (I2C (*Inter-integrated Circuit*) e SPI) e interfaces serial. Estas funcionalidades são permitidas através de saídas fêmea que constam no controlador, às quais se pode ser conectado cabos ou *shields*.

**Tabela 1 – Especificações completas do Arduino UNO[28]**

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by boot-loader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	14g

### 2.3.2.2 RaspberryPi 3b+

O RaspberryPi é um computador de baixo custo e dimensão, com capacidade de interagir com o exterior e que tem sido usado numa vasta quantidade de projetos, inicialmente foi usado para a educação, no entanto a sua pequena dimensão e poder de processamento significativo, começou também a ser utilizado por entusiastas [29].



Figura 2.14 - RaspberryPi 3b+ [28]

O RaspberryPi pode utilizar o sistema operativo Linux assim como o Windows 10, possui quatro saídas USB (*Universal Serial Bus*) embebidas na placa que permite a conexão de rato, teclado, ecrã HDMI (*High Definition Multimedia Interface*), Ethernet, existe a possibilidade de ser expandido através de um USB hub.[27]

Em seguida são expostas as especificações técnicas do mesmo dispositivo.

Tabela 2 - Especificações do RaspberryPi 3b+ [27]

Features	RaspberryPi Pi Model B+
SoC	BCM2835
CPU	ARM11
Operating Freq.	700 MHz
RAM	512 MB SDRAM
GPU	250 MHz Videocore IV
Storage	micro-SD
Ethernet	Yes
Wireless	WiFi and Bluetooth

## 2.4 Comunicação

De modo a se estabelecer a comunicação entre o controlador de aquisição de dados e o RaspberryPi, utilizou-se a comunicação série, mais precisamente o protocolo **UART** (*Universal Asynchronous Receiver Transmitter*). A comunicação **I2C** também foi utilizada na comunicação com o Microcontrolador MPU6050.

### 2.4.1 UART

Este protocolo é maioritariamente utilizado para comunicações a curtas distâncias. Este tipo de comunicação possui um elevado grau de confiabilidade e transmissão de longa distância, logo é bastante utilizado em transferência de dados entre o computador e os seus periféricos. A tecnologia UART permite a comunicação *full-duplex*, isto é, tanto o transmissor como o têm a possibilidade de enviar dados.

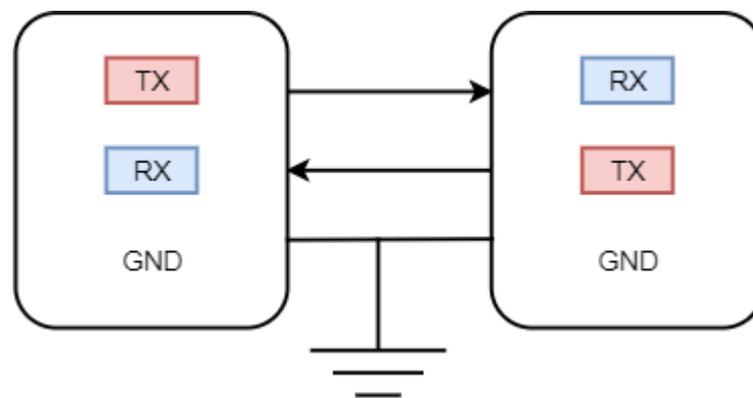


Figura 2.15 - Esquema da Comunicação

A comunicação através de UART necessita apenas de dois sinais (RX e TX) para completar a comunicação *full-duplex*, sendo a linha TX o lado do transmissor e o RX será o recetor [37].

### 2.4.2 I2C

A comunicação I2C é bastante popular e usada em vários dispositivos eletrônicos pois pode ser facilmente implementada em dispositivos que necessitem de comunicação entre um dispositivo *Master* e variados *Slaves*.

A comunicação entre vários dispositivos é possibilitada devido ao facto de cada dispositivo possuir um ID específico, desta forma o dispositivo Master apenas terá que escolher o ID para efetuar a comunicação. Para ser efetuada a comunicação são apenas necessários 2 cabos o SCL(*Serial Clock*) e o SDA (*Serial Data*). O SCL refere-se ao *clock* que sincroniza a transferência de dados entre os dispositivos ligados ao *bus* I2C, este *clock* é gerado pelo dispositivo *Master*. O SDA tem como função transportar a informação [34].

# Arquitetura do Interpretador LGP

## 3

Foi apresentado no capítulo anterior uma descrição do tema da Língua Gestual Portuguesa, bem como os problemas associados à surdez. Foram também estudadas algumas tecnologias que visavam solucionar este problema, com tecnologias distintas. No final foi apresentada uma de solução, a qual será explicitada com maior profundidade neste capítulo.

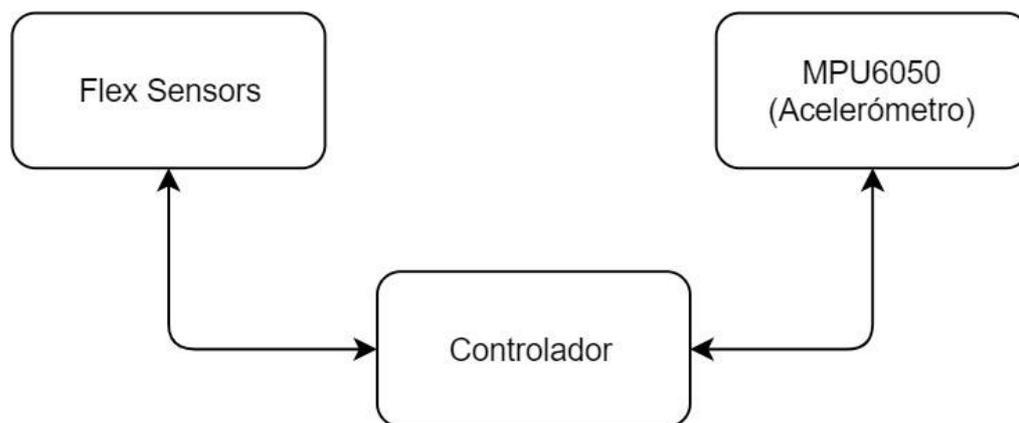
Este capítulo tem como principal objetivo apresentar a estrutura geral do protótipo em questão bem como explicar em maior profundidade os módulos utilizados ao longo do desenvolvimento deste projeto.

### 3.1 Arquitetura do Sistema

A solução utilizada trata-se de um sistema, que ao mais alto nível, permite a interação do utilizador com este mesmo sistema da seguinte forma:

- O **utilizador** irá interagir com o sistema e tem a tarefa de realizar os gestos necessários para gerar o sinal de saída que irá ser lido pelo controlador.
- O **Sistema** tem de reagir mediante os valores dados pelos sensores e fazer o mapeamento destes valores de forma conveniente.
- Por fim o **Sistema Controlado** irá decodificar a informação enviada pelo controlador obtendo o resultado final.

Para realizar a aquisição de sinais derivado dos sensores foi utilizado 1 Arduino nano, 5 *Flex Sensors*, 1 giroscópio *mpu6050* e 5 resistências.



**Figura 3.1 - Sistema de aquisição de dados**

De início pensou-se o Arduino Uno, no entanto através da experiência notou-se que faltariam portas analógicas devido à inserção do giroscópio na equação.

Através desta solução foi possível ser captado o sinal registado por cada dedo e ao mesmo tempo é possibilitado o registo da posição angular do giroscópio, o que irá ser útil na classificação do gesto, como irá ser demonstrado adiante.

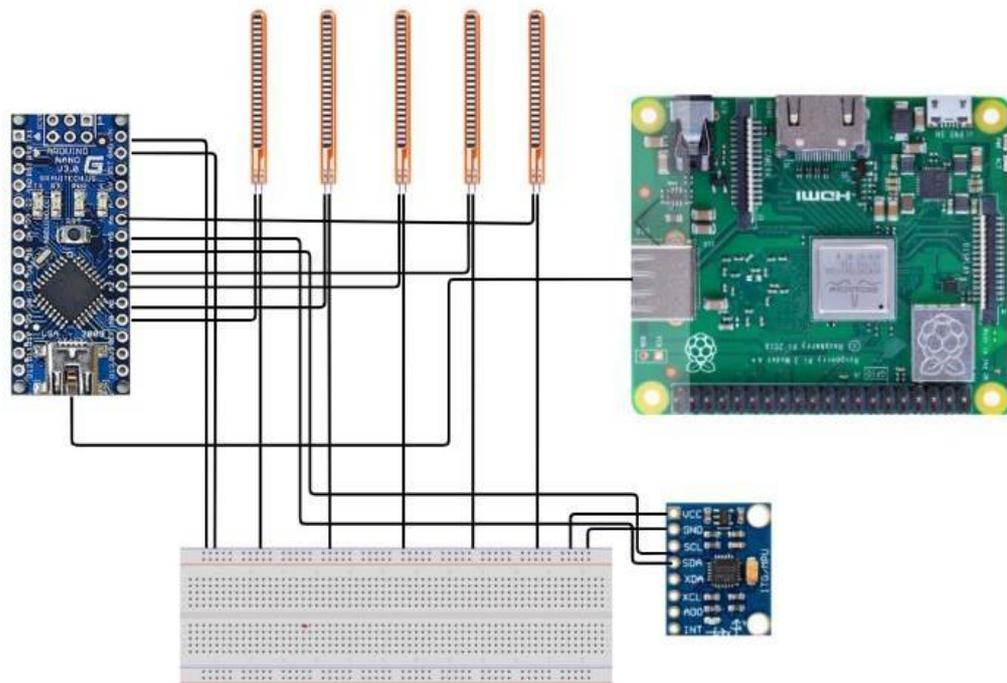


Figura 3.2 – Arquitetura do Sistema

### 3.1.1 Ink Based Flex Sensor

Como foi explicado no capítulo anterior foram utilizados *Flex Sensors* ou *Bend Sensors* para a realização da aquisição de dados, mais precisamente foram usados os *Ink based Flex Sensors*.

O seu funcionamento baseia-se num divisor de tensão formado pelo próprio sensor, numa resistência que é acrescentada e um amplificador operacional AmpOp.

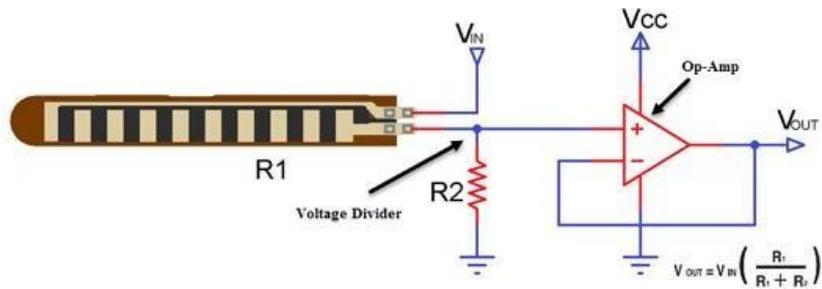


Figura 3.3 - Circuito do *Flex Sensor*

Como podemos verificar na Figura 3.3, existe uma resistência associada ao sensor que variará mediante a quantidade de arqueamento do sensor, esse valor irá ser conjugado a uma resistência fixa, da qual derivará um valor de tensão de saída. Esse valor de tensão será o valor retido pelo controlador.

### 3.1.2 MPU6050

O MPU-6050 é um pequeno controlador que possui 2 tipos de sensores um acelerômetro e um giroscópio de alta precisão. Além destes 2 sensores, tem a seu dispor um recurso chamado DMP (*Digital Motion Processor*), que é responsável por fazer cálculos de elevada complexidade com os sensores que poderão ser úteis para determinar a posição do sensor.

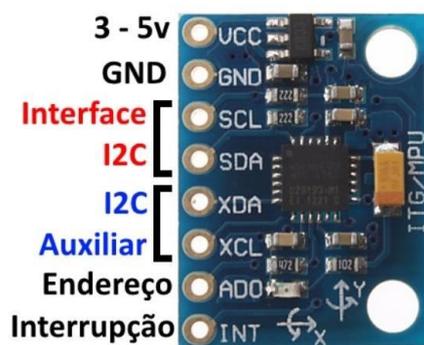


Figura 3.4 - Estrutura MPU6050

A comunicação deste MCU utiliza a interface I2C, por meio dos pinos SCL e SDA do sensor. É possível a conexão dos pinos XDA e XCL a outros dispositivos

I2C de modo a criar um sistema de orientação mais completo. No entanto, não será necessário no âmbito deste protótipo. Relativamente à alimentação do módulo, esta pode variar entre 3 e 5v. Neste caso foi optado pela alimentação a 5v.

### 3.1.3 Sistema completo

Para a comunicação entre o Arduino e o RaspberryPi, que tem como objetivo processar a informação adquirida pelo Arduino, decidiu-se optar pela comunicação por cabo UART, devido à facilidade da configuração com este tipo de comunicação. No entanto, seria útil implementar algum tipo de comunicação sem fio. Na Figura 3., podemos observar o sistema completo e a sua montagem.



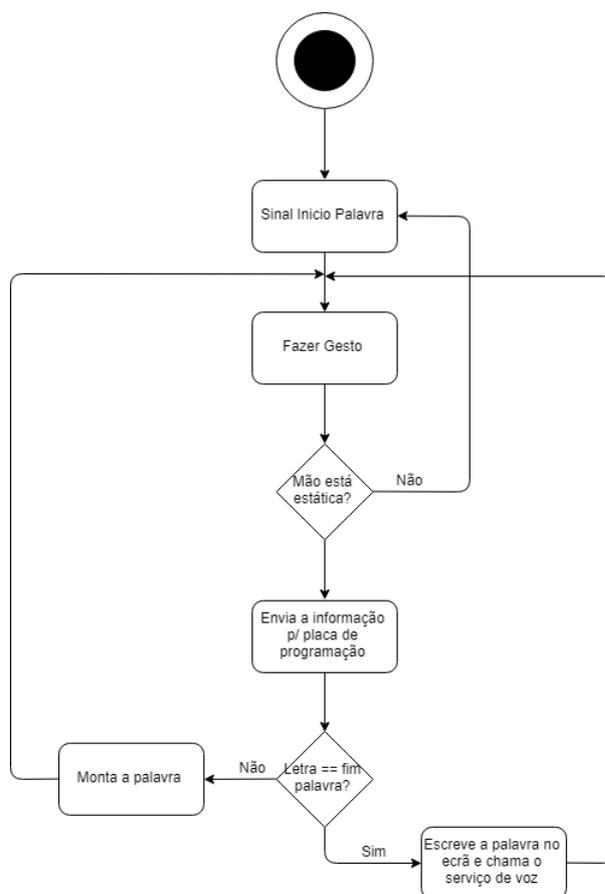
**Figura 3.5 - Modelo Luva tradutora LGP**

Como podemos observar na Figura 3., todos os módulos descritos acima estão conectados e os sensores acoplados a uma luva dentro de várias

membranas de neoprene. Desta forma, o utilizador irá alterar o fluxo de dados apenas com o mexer dos dedos.

### 3.1.4 Fluxograma de funcionamento

Na Figura 3. pode-se observar a sequência de atividades que terão de acontecer de modo a se obter as letras e, consequentemente, a palavra num todo.



**Figura 3.6 – fluxograma de funcionamento do sistema**

Como podemos observar na Figura 3.6, o primeiro estado está descrito como início de palavra, isto porque tem de existir algum sinal de modo a que se saiba que o controlador passará ao estado no qual está pronto para enviar dados, para posteriormente serem processados. Em seguida temos os seguintes passos.

- Primeiramente a pessoa manipula a sua mão de modo a fazer o gesto que permite a escrita da letra que deseja;
- Após a pessoa ter executado o gesto tem de garantir que a mão está estática, de modo a que os dados possam sair do controlador e passar a uma fase de análise de dados. Caso ela não esteja estática o sinal não poderá ser enviado;
- De seguida o sinal é enviado para a placa de programação onde é verificado se a letra é equivalente ao sinal de fim de palavra;
- Caso não seja igual ao fim de palavra, a letra irá ser colocada na *string* de *buffer*, caso a letra for igual a fim de palavra, ela irá ser enviada para o serviço de voz para ser reproduzida.

# Implementação do protótipo



Neste capítulo, será apresentada a descrição do sistema, assim como o protótipo do projeto. Encontra-se também detalhado o funcionamento de todos os sensores e módulos utilizados. Estarão também presentes os diagramas de montagens do circuito.

## 4.1 Montagem de arquitetura

Na Figura 4.1 é possível se observar toda a montagem relativa ao controlador, incluindo os sensores utilizados, o microcontrolador MPU6050, bem como a montagem de *resistência*.

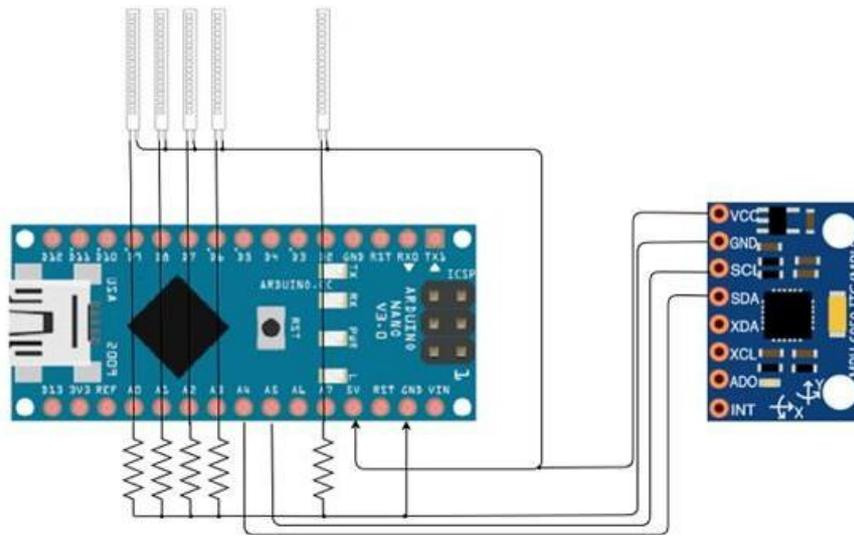


Figura 4.1 - Montagem completa

## 4.2 Escolha de resistências

Para a escolha de resistências primeiramente utilizou-se um multímetro para calcular as respectivas resistências enquanto o sensor está direito e quando está dobrado totalmente.

R2	30000	150000	Vrange
10000	3,75	4,6875	0,9375
11000	3,658537	4,658385	0,999849
12000	3,571429	4,62963	1,058201
13000	3,488372	4,601227	1,112855
14000	3,409091	4,573171	1,16408
15000	3,333333	4,545455	1,212121
16000	3,26087	4,518072	1,257203
17000	3,191489	4,491018	1,299529
93000	1,219512	3,08642	1,866908
94000	1,209677	3,07377	1,864093
95000	1,2	3,061224	1,861224
96000	1,190476	3,04878	1,858304
97000	1,181102	3,036437	1,855335
98000	1,171875	3,024194	1,852319
99000	1,162791	3,012048	1,849257
100000	1,153846	3	1,846154
101000	1,145038	2,988048	1,84301
102000	1,136364	2,97619	1,839827
103000	1,12782	2,964427	1,836607
104000	1,119403	2,952756	1,833353

**Figura 4.2 - tabela de valores de resistência para o dedo mindinho**

A tabela da Figura 4.2 descreve parcialmente os valores de resistência que foram avaliados para uma tensão de entrada de 5V. O primeiro número o 30000 em cima corresponde ao valor de resistência inferior que o sensor pode registrar enquanto que o superior, o 150000 é o valor superior, todos estes valores estão registados em Ohms.

A cada valor de tensão de saída foi aplicada a seguinte fórmula para calcular a tensão de saída:

$$V_{out} = V_{in} * \frac{R_{flexsensor}}{R_{flexsensor} + R_{resistência}} \quad (4.1)$$

Após a obtenção de todos os valores de tensão de saída para uma dada resistência, o objetivo é verificar qual o valor de resistência que permite uma

maior variação de valores, com o intuito de que os valores sejam mais fáceis de caracterizar. Podemos observar na imagem a cima que o valor ótimo seria cerca de 94K $\Omega$ ., no entanto, não tendo encontrado esse valor de resistência optou-se o uso de uma resistência de 100 k $\Omega$ .

Depois de calculadas todas as resistências para cada um dos dedos, obteve-se então os seguintes dados:

- Para dedo polegar e mindinho – R = 100 k $\Omega$
- Para dedo indicador, médio e anelar – R = 30 k $\Omega$

De notar que o sensor para dedo indicador, médio e anelar tem algumas diferenças no que toca aos seus valores de resistência, assim como para o mindinho e polegar.

### 4.3 Aquisição de dados

Nesta fase de aquisição de dados é necessário que estes sejam enviados pelo controlador para serem tratados e, mais tarde, processados e avaliados. Procedeu-se então ao tratamento de dados relativos aos *Flex Sensors* e aos valores do acelerómetro do MPU6050.

- **Flex Sensors**

De modo a que os valores possam ser tratados de forma mais eficiente e prática, os valores lidos pelo controlador passarão a ser descritos por valores distintos dos originais. Ou seja, serão caracterizados consoante o seu arqueamento, maior ou menor, através de um mapeamento de valores, para serem enviados de seguida para o RaspberryPi.

### 4.3.2 MPU 6050

Como já foi salientado no capítulo anterior, foi utilizado o controlador MPU6050 que, é na sua essência, um sensor de deteção de movimento, também conhecido como o primeiro dispositivo no mundo capaz de detetar movimento a 6 dimensões.

Para a sua utilização, primeiramente foi iniciada a transmissão usando o seguinte endereço I2C.

PARAMETER	CONDITIONS	MIN	TYP	MAX	Units	Notes
<b>SERIAL INTERFACE</b>						
SPI Operating Frequency, All Registers Read/Write	MPU-6000 only, Low Speed Characterization		100 ±10%			kHz
	MPU-6000 only, High Speed Characterization		1 ±10%			MHz
	MPU-6000 only		20 ±10%			MHz
SPI Operating Frequency, Sensor and Interrupt Registers Read Only	All registers, Fast-mode			400		kHz
	All registers, Standard-mode			100		kHz
<b>I<sup>2</sup>C ADDRESS</b>	AD0 = 0		1101000			
	AD0 = 1		1101001			
<b>DIGITAL INPUTS (SDI/SDA, AD0, SCLK/SCL, FSYNC, /CS, CLKIN)</b>						
V <sub>IH</sub> , High Level Input Voltage	MPU-6000	0.7*VDD				V
	MPU-6050	0.7*VLOGIC				V
V <sub>IL</sub> , Low Level Input Voltage	MPU-6000			0.3*VDD		V
	MPU-6050			0.3*VLOGIC		V
C <sub>i</sub> , Input Capacitance			< 5			pF
<b>DIGITAL OUTPUT (SDO, INT)</b>						
V <sub>OH</sub> , High Level Output Voltage	R <sub>LOAD</sub> =1MΩ; MPU-6000	0.9*VDD				V
	R <sub>LOAD</sub> =1MΩ; MPU-6050	0.9*VLOGIC				V
V <sub>OL1</sub> , LOW-Level Output Voltage	R <sub>LOAD</sub> =1MΩ; MPU-6000			0.1*VDD		V
	R <sub>LOAD</sub> =1MΩ; MPU-6050			0.1*VLOGIC		V
V <sub>OLINT1</sub> , INT Low-Level Output Voltage	OPEN=1, 0.3mA sink Current			0.1		V
Output Leakage Current	OPEN=1		100			nA
t <sub>INT</sub> , INT Pulse Width	LATCH_INT_EN=0		50			μs

Figura 4.3 - I2C adress MPU6050

Como podemos na Figura 4.3 ver o último bit do registo pode ser 1 ou 0. Isto deve-se ao facto de ser possibilitado a utilização de um segundo MPU, caso seja desejado. De fábrica o pino ADO vem conectado a *Ground* via software, no entanto se se quiser utilizar um segundo dispositivo o seu endereço I2C seria 1101001, utilizando então este pino ADO.

De seguida na Figura 4.4, devido ao facto de o MPU vir em modo *sleep-mode* de fábrica é necessário que este seja desativado, isto é realizado utilizando o seguinte registo:

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6B	107	DEVICE_RESET	SLEEP	CYCLE	-	TEMP_DIS	CLKSEL[2:0]		

Figura 4.4 - Registo 0x6B

**Acelerómetro** – Foi utilizado o acelerómetro para ser calculado o ângulo no qual se encontra o sensor. A razão pela qual não se utilizou o giroscópio foi porque as informações retiradas por ele são relativas à variação angular e não de um ângulo estático. Para se aceder ao registo do acelerómetro foi utilizado o registo 0x1C e foi escolhido uma sensibilidade de 2g o que permite obter valores de output do acelerómetro de -1 a 1.

Nesta secção, pode-se observar o impacto na aceleração quando é variada a posição do controlador relativamente ao espaço. No fundo observa-se o impacto das forças aplicadas no controlador.

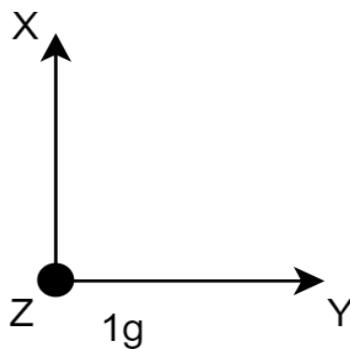
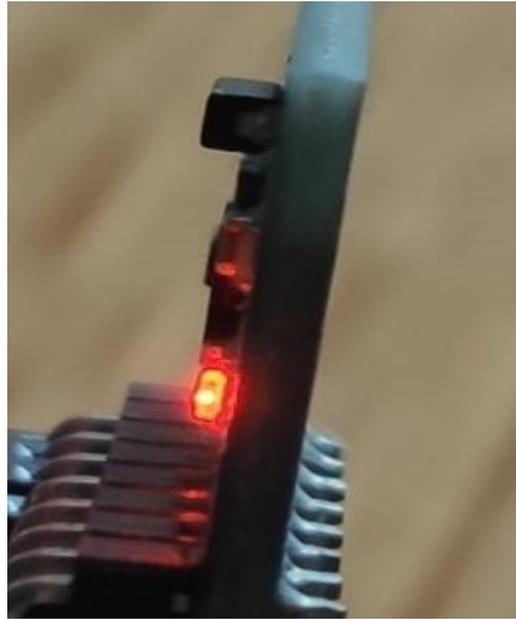


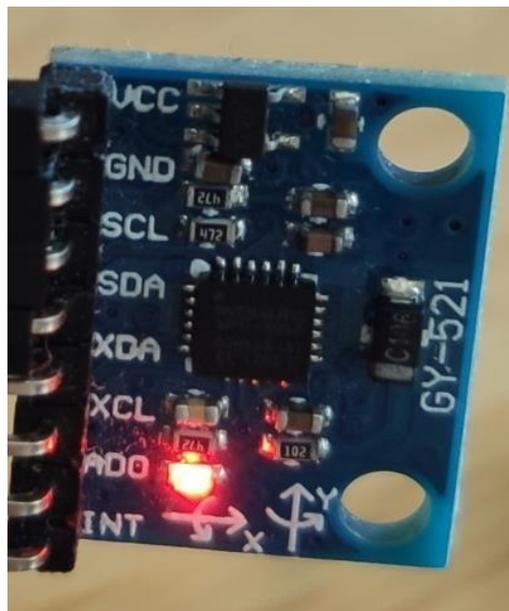
Figura 4.5 - Forças aplicadas quando o sensor está numa posição plana

Como podemos observar na Figura 4.5, quando o controlador se encontra na posição horizontal a única força aplicada é a força no eixo Z. Nos restantes eixos não existem forças a serem aplicadas. Tem-se então o valor do acelerómetro nos eixos X, Y e Z respetivamente (0, 0, 1).



**Figura 4.6 - Mpu inclinado sobre o eixo dos Y**

Neste caso (Figura 4.6), o controlador encontra-se na posição vertical o que faz com que a força aplicada anteriormente, quando o controlador estava na vertical, deixe de existir e passe a ser antes uma força sobre o eixo dos Y ao invés de Z, ou seja, neste caso teríamos os eixos X, Y e Z a ser respetivamente (0, 1, 0).



**Figura 4.7 - mpu inclinado sobre o eixo dos X**

Aqui (Figura 4.7) como no caso anterior temos uma inclinação sobre o eixo dos X, ou seja, os valores X, Y e Z serão respetivamente (1,0,0).

**Processamento da informação** – Após concluída a configuração do mpu é necessário que a informação do acelerómetro seja retirada. Para tal, irá se aceder ao registo 0x3B até ao registo 0x40, registos estes, de onde se extrai a informação do acelerómetro. É feito então um pedido para se obter os bytes correspondentes a cada informação. Cada valor de aceleração, em cada eixo, vem incluído em 2 bytes, ou seja, os registos 3B e 3C possuem a tarefa de armazenarem os valores de aceleração no eixo X, os registos 3D e 3E monitorizam a aceleração no eixo dos Y e 3F e 40, a aceleração no eixo dos Z. Tendo sido obtido o valor retornado pelos bytes dos registos poderemos então olhar para a seguinte tabela 3:

Tabela 3 - Tabela de sensibilidades

AFS_SEL	Full Scale Range	LSB Sensitivity
0	±2g	16384 LSB/g
1	±4g	8192 LSB/g
2	±8g	4096 LSB/g
3	±16g	2048 LSB/g

Como foi dito anteriormente, foi escolhida a sensibilidade de  $\pm 2g$  (coeficiente gravitacional). À direita podemos observar um valor correspondente de sensibilidade LSB (*Least Significant Byte*) de 16384 LSB/g. E tendo em conta que o valor que irá ser lido pela consola está em “g”, o valor lido pelo registo terá de ser em “g”. Para efetuar a conversão, e para efeitos de explicação, vamos supor que o registo 0x3B e 0x3C retornam um valor de 15000. Para determinarmos o valor correspondente em “g”, teria de ser feito  $15000/16384$  e então obter o valor em “g” que neste caso seria 0,92.

### 4.3.3 Envio de informação

Para enviar informação relativa aos *Flex Sensors* e giroscópio, primeiramente efetuou-se um mapeamento dos mesmos valores para novos valores, de modo a ser a simplificar os dados enviados e a sua própria leitura. Para tal, teve se em consideração o propósito dos próprios valores, ou seja, qual seria o grau de complexidade que deveria ser utilizado para conseguir distinguir os dados, neste caso, o objetivo é distinguir cada letra.

Outro aspeto que também se teve em consideração foi a sensibilidade de cada dedo. Como podemos verificar, por exemplo, o dedo anelar terá uma sensibilidade distinta do dedo médio ou indicador na maioria das pessoas, por isso, a gama de valores lidos do dedo anelar deverá ser distinta dos demais.

Tabela 4 - Mapeamento de valores consoante o ângulo dos flex sensors

Ângulo	0	30	60	90	90+
Dedos					
Polegar	0	1	2	2	2
Indicador	0	1	2	3	4
médio	0	1	2	3	4
Anelar	0	1	2	2	2
Mindinho	0	1	2	2	2

Como podemos observar na Tabela 4, foram atribuídos valores de 0-4 aos sensores que se considerou terem uma necessidade de maior variabilidade para a distinção de cada letra. E foi atribuída um mapeamento de 0-2 para os que possuem um menor grau de sensibilidade.

0	4	4	2	2	0
0	4	4	2	2	0
0	4	4	2	2	0
0	4	4	2	2	0
0	4	4	2	2	0
0	4	4	2	2	0

**Figura 4.8 - Valores recebidos executando a forma da letra B**

Como podemos observar, foi efetuado então o gesto da letra B, para teste, e foram atribuídos os valores da Figura 4.8. A razão pela qual conseguimos identificar 6 linhas é devido ao facto de, desta maneira, poder se ter um maior grau de confiança, comparativamente ao envio dos dados numa só iteração. Cada linha acima representada representa um ciclo de envio de dados, isto é, podemos observar cada valor de cada *Flex Sensor*, ou seja, os valores do dedo polegar até ao mindinho.

O mesmo processo foi efetuado para o envio de dados correspondentes ao giroscópio, ou seja, foram convertidos os dados enviados pelos registos do microcontrolador e mapeados para valores de mais fácil leitura.

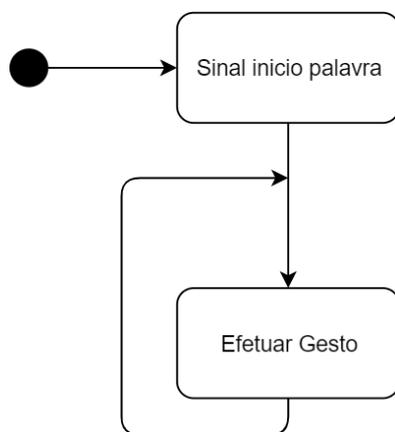
RotX	RotY	RotZ
0	40	0
0	40	0
0	40	0
0	40	0
0	40	0
0	40	0

**Figura 4.9 - Valores do giroscópio relativos à letra "B"**

Como podemos verificar na Figura 4.9, conseguimos identificar 3 valores distintos recebidos, que significam da esquerda para a direita, os valores de rotação nos eixos X, Y e Z. A razão pela qual eles tomarem valores distintos de -1, 0 e 1 como explicado a cima, é simplesmente para facilitar a distinção dos

mesmos, ou seja, para tornar mais vinculada a diferença entre cada posição do controlador, no entanto, irá se estudar este tema em maior pormenor mais adiante nesta dissertação.

Tendo sido então explicado como é efetuado o processo de aquisição de dados tanto dos *Flex Sensor* bem como do acelerómetro, podemos verificar no diagrama da 4.12 como funciona o processo de aquisição de dados na prática



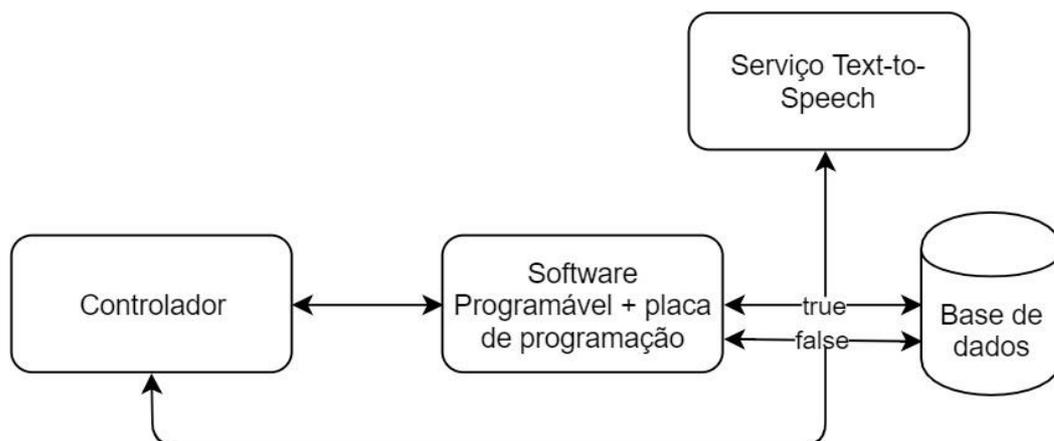
**Figura 4.10 - Diagrama comportamental relativo à aquisição de dados**

Para inicializar o processo de aquisição de dados é necessário haver algum sinal que o distinga (Figura 4.10). Neste caso foi definido o próprio sinal de início de palavra. De seguida, é esperado que o utilizador realize o gesto que pretende, de forma a posteriormente obter a letra correspondente. Após concluído o envio de dados será efetuado um novo ciclo de aquisição de dados, no entanto, o utilizador não necessita de efetuar o gesto de início de palavra novamente.

#### **4.4 Tratamento de Dados**

No que toca ao tratamento de dados, como já foi explicitado, utilizou-se o RaspberryPi para realizar todo o tipo de operações com os dados obtidos pelo controlador. Foi necessário o desenvolvimento de um programa em Python para

efetuar todo o tipo de tratamento de dados, com o auxílio de uma base de dados que contem informação acerca de todas as letras do alfabeto e os correspondentes valores de arqueamento sensores e do acelerómetro do giroscópio. Estes valores encontram-se no mesmo formato dos dados enviados pelo controlador.



**Figura 4.11 - Interação Arduino RaspberryPi**

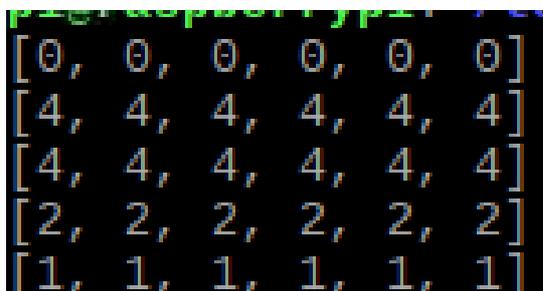
Como se verifica na Figura 4.11, podemos ver o esquema geral de funcionamento deste projeto, onde se verifica a interação entre os dados enviados pelo controlador a base de dados e como estado final o envio da palavra para o serviço Text-to-speech.

- **Tratamento de dados via Python**

Nesta fase, o principal objetivo era fazer um programa que fosse capaz de receber os valores enviados pelo controlador, com o mínimo de erros possível, interpretando-os de forma a se obter o caracter que é pretendido e, por fim, determinar uma palavra construída através dos vários caracteres.

Numa fase inicial, testou-se a comunicação entre o RaspberryPi e o Arduino enviando simples valores, para tal apenas se definiu no código Python do

RaspberryPi, a porta série utilizada pelo Arduino e a frequência de *baudrate* que tem de ser a que foi previamente definida no Arduino.



```
[0, 0, 0, 0, 0, 0]
[4, 4, 4, 4, 4, 4]
[4, 4, 4, 4, 4, 4]
[2, 2, 2, 2, 2, 2]
[1, 1, 1, 1, 1, 1]
```

Figura 4.12 - Valores dos flex sensores no RaspberryPi enviados pelo Arduino

Como podemos observar, os dados são recebidos pelo RaspberryPi com sucesso Figura 4.12. Para tal utilizou-se um ciclo que irá ler infinitamente toda a informação enviada pelo Arduino após o sinal de início de palavra e através de uma técnica de *split*, presente nas bibliotecas de Python, que neste caso, irá separar os dados dividindo-os em valores independentes a cada dedo. A razão pela qual podemos observar vários valores iguais é devido ao facto de se efetuar 6 ciclos de envio de dados de modo a obter-se um maior valor de confiança comparando ao método de apenas realizar um ciclo, visto que durante o envio de dados a pessoa pode mexer ligeiramente a mão e alterar o fluxo de dados. Como podemos observar os valores recebidos respeitam a configuração explicada anteriormente.

Após obter-se os valores dos sensores e do acelerómetro a ideia foi somar todos os valores que sejam do respetivo sensor, ou seja, somam-se os valores de cada dedo em todas as iterações para posteriormente se efetuar uma função de média e assim obter um valor aproximado à realidade.

- **Base de Dados**

Foi utilizado o *Sqlite* como plataforma de base de dados, devido ao facto de ser uma plataforma leve e de relativa fácil utilização. A sua principal utilidade é a de armazenar todos os valores de arqueamento correspondentes a cada letra,

no formato correto. Esta base de dados apenas serve para leitura, isto é, não há nenhuma altura onde haja escrita para a base de dados.

ID	Letra	V1	V2	V3	V4	V5	Rot1	Rot2	Rot3
2	b	0	4	3	2	2	0	40	0
3	c	0	3	3	2	2	0	0	40
4	d	0	0	0	0	0	0	0	20
5	e	0	3	4	2	2	40	0	0
6	f	0	3	0	0	0	40	0	0
9	i	0	4	4	2	0	40	0	0
10	j	1	0	0	0	0	40	0	0
11	k	0	0	0	2	2	40	0	0
12	l	0	0	4	2	2	40	0	0
13	m	2	0	0	0	2	20	0	0
14	n	2	0	0	2	2	20	0	0
15	o	1	3	3	2	1	0	0	40
16	p	0	4	0	0	0	0	0	20
17	q	0	4	4	2	2	0	20	0
18	r	0	3	0	2	2	40	0	0
19	s	2	4	4	2	2	40	0	0
20	t	0	0	4	2	2	0	40	0
21	u	1	0	0	2	2	40	0	0
22	v	0	0	0	2	2	40	0	0
23	w	2	0	0	0	2	40	0	0
24	x	1	0	2	2	2	40	0	0
25	y	0	4	4	2	0	40	0	0
26	z	1	3	4	2	2	40	0	0
27	/	2	4	4	2	2	0	0	40

Figura 4.13 - Base de dados

Encontram-se todas as letras do alfabeto português na Figura 4.13, onde podemos observar os valores atribuídos, tendo em conta todo o mapeamento efetuado.

Na seguinte secção, poderemos entender melhor como foi feita a atribuição da letra mediante os valores recebidos pelo controlador e como se relacionam com a base de dados.

- **Obtenção de dados relativos à letra**

Foi observado que se teria de obter os valores médios para cada dedo de forma a obter um maior grau de confiança. Posto isto efetuou-se uma query à base de dados, cujo principal objetivo é entender qual seria o carácter presente na base de dados que teria maior semelhança ao enviado pelo controlador. Assim, utilizou-se a função absoluto, efetuando a diferença de valores entre cada sensor correspondente à média e os valores presentes na base de dados e posteriormente retornando o que tivesse menor diferença (carácter escolhido).

Letra	Di - Df
('b', 1.0)	
('t', 4.0)	
('k', 8.0)	
('q', 20.0)	
('p', 68.0)	
('n', 70.0)	
('d', 72.0)	
('m', 72.0)	
('e', 81.0)	
('g', 81.0)	
('c', 82.0)	
('i', 82.0)	
('s', 82.0)	
('y', 82.0)	
('z', 82.0)	
('/', 82.0)	
('l', 84.0)	
('o', 84.0)	
('a', 85.0)	
('r', 85.0)	
('x', 87.0)	
('v', 88.0)	
('f', 89.0)	
('h', 89.0)	
('u', 89.0)	
('w', 92.0)	
('j', 93.0)	

Di - Valor absoluto
médio dos sensores
da base de dados
Df - Valor absoluto
médio dos sensores
obtido pelo
controlador

**Figura 4.14 - Output com distâncias absolutas relativo à informação enviada**

Através dos valores enviados pelo controlador e valores presentes na base de dados, é construída a tabela da Figura 4.14 onde podemos claramente ver o módulo da distância entre eles. Esta tabela permitirá decidir qual é a letra, cujos valores dos sensores são os mais aproximados à realidade, o que nos permite escolher a letra a ser enviada. O passo seguinte é retirar a letra que consta no topo da lista visto que a tabela está organizada da distância menor para a maior, para posteriormente colocar essa letra numa lista que irá construir, por fim, a palavra.

- **Corretor de palavras pySpellchecker**

Devido ao facto do presente sistema possuir algumas falhas, assim como letras que são complicadas de ser obtidas, devido à complexidade do gesto, optou-se pela utilização de uma biblioteca do Python a PySpellchecker, que permite a correção ortográfica de palavras. Este corretor utiliza a distância de Levenshtein [37], que permite obter permutações a uma certa distância da palavra original. Posteriormente são comparadas todas as permutações, assim como, inserções, remoções, substituições e transposições de palavras conhecidas numa lista de frequência. É de notar, que as palavras que constem na lista de frequência com maior regularidade são mais propícias a serem os resultados de palavra correta [37].

```

"abacates": 50,
"abacaxi": 152,
"abada": 92,
"abaddon": 131,
"abade": 266,
"abadia": 316,
"abafado": 269,
"abafar": 299,
"abaixa": 660,
"abaixada": 178,
"abaixado": 243,
"abaixados": 179,
"abaixar": 256,
"abaixe": 772,
"abaixem": 306,
"abaixo": 13768,
"abajur": 130,
"abala": 63,
" . . . . . " . . .

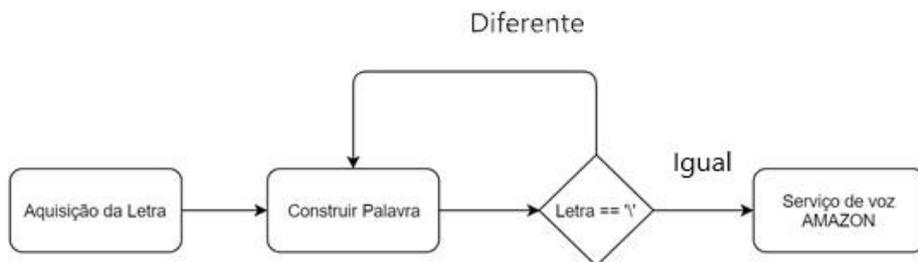
```

**Figura 4.15 - Exemplo de Lista de frequência**

Pode-se observar na Figura 4.15 uma porção da lista de frequência utilizada. Para entender melhor um pouco do conceito da distância de Levenshtein, vamos supor que queríamos escrever a palavra “abaixo”, acima representada, no entanto foi escrito “abalxado” pelo protótipo, a distância de Levenshtein dita o número de edições, ou seja, inserções, remoções ou alterações que são possíveis efetuar, esta biblioteca permite a utilização de distância de 1 ou de 2, neste caso seria apenas necessário utilizar a distância de 1.

#### **4.5 Diagrama Comportamental funcionamento do protótipo**

Na Figura 4.16, podemos verificar o funcionamento do protótipo a nível de obtenção de dados, bem como o envio dos mesmos, para o serviço de fala através do seguinte diagrama comportamental:



**Figura 4.16 - Funcionamento do protótipo**

Como podemos identificar na Figura 4.16, a construção da palavra é feita letra a letra, isto é, após ser enviada a informação e reconhecida pela interação com o programa é inserida a letra numa palavra de buffer que irá ser preenchida à medida que é escrita a palavra. A cada iteração é verificado se o sinal de entrada é equivalente ao sinal da letra '\', equivalente ao sinal fim de palavra. Quando é encontrado o tal sinal de palavra, esta é enviada para uma função que irá fazer a reprodução de som.

Para a reprodução de som, foi utilizada uma biblioteca de Python chamada Pygame que é uma Framework de desenvolvimento de jogos que possui uma funcionalidade de reprodução de áudio. O seu propósito será reproduzir o ficheiro de áudio criado pelo serviço Amazon. Neste capítulo deu-se a entender como funciona este protótipo na prática, no capítulo seguinte ir-se-á apresentar os testes realizados à arquitetura.



# Testes e Resultados



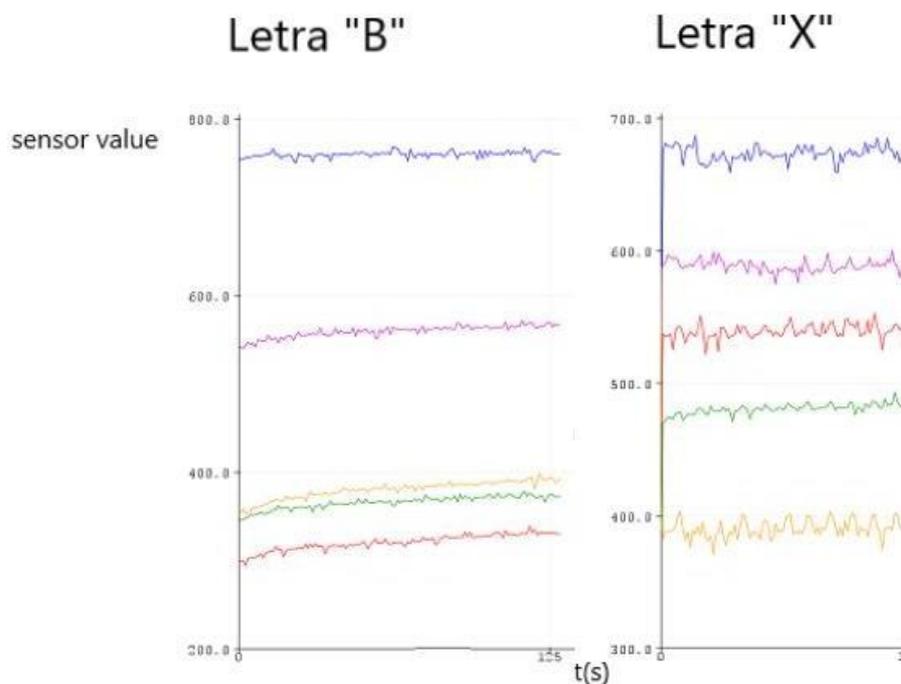
Neste capítulo será introduzida uma série de testes experimentais relativos ao protótipo do sistema. Será também efetuado um teste de velocidade na aquisição e obtenção de dados finais, bem como a eficácia do sistema de correção de palavra.

## 5.1 Testes de aquisição de dados

Nesta secção pretende-se efetuar e apresentar alguns testes do sistema recorrendo-se ao protótipo.

Num **primeiro teste** foi efetuado uma letra que se considerou bastante simples, a letra “B”. No **segundo teste**, uma letra um pouco mais complexa o “X”.

Na Figura 5.1 – Aquisição da letra “B” e “X”, poderemos encontrar os resultados retirados da aquisição de dados feita através da letra “B” e “X”, onde cada linha representa a variação dos valores respetivos a cada sensor



**Figura 5.1 – Aquisição da letra “B” e “X”**

Primeiramente é de notar que os valores representados derivam da tensão de saída regulada pela resistência variável do *Flex Sensor* e que a razão pela qual os valores são os apresentados, deve-se ao facto da tensão estar representada em 10 bits neste caso os 5V representariam 1023.

Na Figura 5.1 – Aquisição da letra “B” e “X” podemos observar uma maior oscilação de dados no gráfico da direita em comparação com o gráfico da esquerda. Isto deve-se ao facto de nos testes realizados, ter se notado uma maior dificuldade a realizar a letra “X” comparativamente à letra “B”, ou seja, foi notado uma maior instabilidade da mão na realização do gesto. Esta instabilidade tem como consequência um maior tempo necessário para aquisição da letra.

## 5.2 Tratamento de dados

Nesta secção, ir-se-á se avaliar a obtenção do caracter mediante os dados enviados pelo controlador, irão também ser efetuados testes de rapidez do protótipo, nomeadamente o tempo que é necessário para a formação de um caracter. Por último será também efetuado um teste que serve para avaliar a eficácia da biblioteca de Python utilizada Pyspellchecker.

### 5.2.1 Aquisição de letras e teste de velocidade

Na tabela 5, poderemos observar o tempo demorado para a obtenção das letras de teste:

**Tabela 5 - Teste aquisição de letras**

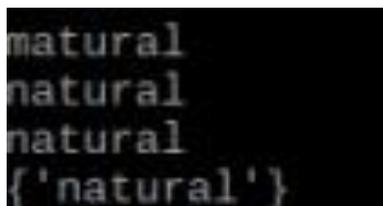
Letra	$\Delta t$ (segundos)
a	5.545
c	9.271
d	5.050
q	13.405
p	5.143

Foram feitos vários testes nestas 5 letras, pois considerou-se que existem algumas com uma maior dificuldade de execução que outras. A dificuldade encontra-se na execução do gesto e dificuldade em manter a mão firme nessas condições. Obviamente que uma pessoa que tiver o hábito de usar a língua gestual portuguesa deverá em teoria obter resultados um pouco superiores.

### 5.2.2 Teste à utilização da biblioteca PySpellchecker

Neste teste o objetivo é realizar uma palavra que esteja escrita com algum tipo de erro ortográfico e verificar se através da utilização da biblioteca

PySpellChecker, a palavra seria corrigida. Neste caso foi utilizada a distância de Levenshtein igual a 1, ou seja, apenas uma alteração à palavra para a palavra supostamente correta.

A terminal window with a black background and white text. The text is displayed on four lines: 'matural', 'natural', 'natural', and {'natural'}.

**Figura 5.2 - Obtenção da palavra na linha de comandos**

Como podemos ver na Figura 5.2, foi propositadamente escrito a palavra “natural” como “matural”, de modo a que se verifique se esta palavra seria corrigida corretamente. Como se pode verificar na imagem retirada da linha de comandos, esta palavra foi corrigida com sucesso.

De maneira geral podemos observar que os resultados foram satisfatórios tanto avaliando parâmetros de rapidez de aquisição de dados, tanto como confiabilidade na obtenção de resultados.

# Conclusões e trabalhos futuros



Neste último capítulo desta dissertação de mestrado são apresentadas algumas considerações relativas ao projeto desenvolvido. Serão também apresentados os trabalhos futuros que poderão ser acrescentados a este protótipo de forma a cobrir algumas necessidades essenciais.

## 6.1 Conclusões

O *Tech4good*, que é o tema principal desta tese de mestrado, e como o nome indica, refere-se a um tipo de tecnologia que pretende resolver algum problema existente relativo à humanidade ou com o nosso ecossistema. Posto isto, foi proposto então o desenvolvimento de um protótipo, que auxiliasse os surdos na comunicação com o exterior.

Com o finalizar do projeto proposto nesta dissertação, tive a oportunidade de explorar a utilização de diversos módulos e entender como poderia captar informação dos mesmos e realizar a interpretação de gestos.

Neste caso em termos de hardware utilizou-se principalmente os *Flex Sensors* e um giroscópio agregados ao Arduino, de modo a realizar o levantamento completo de informações com o intuito de interpretar o gesto, por último, um RaspberryPi para efetuar o tratamento de dados.

Analisando os resultados finais do protótipo, concluímos que de um modo geral que os resultados foram bastante positivos, ou seja, que na maioria dos casos foi feito o reconhecimento de cada letra com sucesso, claramente algumas

letras serão mais complicadas de se obter resultados, devido a posições um pouco desconfortáveis.

Este sistema possui algumas limitações as quais serão explicitadas de seguida.

## **6.2 Trabalhos futuros**

Tendo como base a arquitetura implementada neste projeto, a principal alteração a ser feita, seria a implementação de algum tipo de reconhecimento de imagem, isto devido ao facto de que na língua gestual, na maioria das vezes comunica-se através de gestos, ou seja, a maioria das palavras é comunicada através de um único gesto, o que vai contra o princípio da implementação efetuada nesta dissertação. A ideia seria então ter uma base de dados ainda mais alargada que a existente e proceder-se à avaliação da imagem determinando assim a palavra em questão.

Outra alteração também bastante pertinente, seria a implementação de algum tipo de tecnologia de comunicação sem fio, assim como o Bluetooth, entre o Arduino e RaspberryPi, o que levaria a um aumento de portabilidade do protótipo. Por fim ainda no que toca à comunicação, seria bastante interessante também o desenvolvimento de uma aplicação para telemóvel que pudesse substituir o RaspberryPi. A aplicação poderia ser também desenvolvida em Python. Isto seria uma mais valia, pois simplificaria bastante o sistema a nível de custos e praticidade, pois hoje em dia quase toda a gente possui um smartphone.



## Referências

- [1] “Amazon Polly.” [Online]. Available: <https://aws.amazon.com/pt/polly/>. [Accessed: 17-Jan-2019].
- [2] “Cloud Text-to-Speech: síntese de fala | Cloud Text-to-Speech API | Google Cloud.” [Online]. Available: <https://cloud.google.com/text-to-speech/>. [Accessed: 17-Jan-2019].
- [3] S. Podder, S. K. Singh, P. Tanguturi, P. Roy, C. Gueret, and N. Dubash, “Tech4Good: Collectively scaling up social transformation”, Accenture Research Labs India, 2018.
- [4] Jovita Aranha “This AI-Enhanced App Is Letting Visually Impaired ‘See’ like Never Before.” [Online]. 2018 Available: <https://www.thebetter-india.com/140639/artificial-intelligence-visually-impaired-drishti/>. [Accessed: 17-Jan-2019].
- [5] M. Martins, M. Morgado, P. Estanqueiro, and F. Cavaca, “Programa curricular de língua gestual portuguesa ensino secundário.”, Ministério da Educação, *Universidade Católica Editora, novembro de 2012* (M. Martins, 2006)
- [6] E. A. Lee, “Cyber Physical Systems: Design Challenges,” in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, 2008, pp. 363–369.
- [7] F. Shrouf, J. Ordieres, and G. Miragliotta, “Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm,” in *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, 2014, pp. 697–701.

- [8] I. Lee and K. Lee, “The Internet of Things (IoT): Applications, investments, and challenges for enterprises,” *Bus. Horiz.*, vol. 58, no. 4, pp. 431–440, Jul. 2015.
- [9] R. M. Lima and C. S. Pais Tavares Ferreira Colaço, *Falantes Conscientes, Leitores Competentes*, no. 1. Escola Superior de Educação do Instituto Politécnico de Coimbra, 2009. Available: [dialnet.unirioja.es/servlet/articulo?codigo=3398317](http://dialnet.unirioja.es/servlet/articulo?codigo=3398317)
- [10] W. C. Stokoe, “Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf,” *J. Deaf Stud. Deaf Educ.*, vol. 10, no. 1, pp. 3–37, Jan. 2005.
- [11] Luísa Peixoto “Língua Gestual Portuguesa | Pelo Gesto e pelo Som, comunicar entre surdos e ouvintes-Vila Nova.” [Online]. Available: <https://vilanovaonline.pt/2017/12/10/lgp-pelo-gesto-pelo-som-comunicacao-surdos-ouvintes-luisa-peixoto/> [Accessed: 12-Jan-2019].
- [12] “Learn Sign Language Online (ASL 1 - Unit 3) - Start ASL.” [Online]. Available: <https://www.startasl.com/learn-sign-language-online-asl1>. [Accessed: 12-Jan-2019].
- [13] Luísa M. S. Duarte Freitas. “Elaboração e Avaliação de um Guia Prático para o Ensino da LGP como Segunda Língua”. Coimbra: [s.n.], 2017. Tese de doutoramento. Available: <http://hdl.handle.net/10316/29600>
- [14] J. Fellingner, D. Holzinger, J. Fellingner, D. Holzinger, and R. Pollard, “Mental health of deaf people”, *The Lancet*, vol 379 17 March 2012, Pages 1037-1044, ISSN 0140-6736, Available: [https://doi.org/10.1016/S0140-6736\(11\)61143-4](https://doi.org/10.1016/S0140-6736(11)61143-4) Available: <http://www.sciencedirect.com/science/article/pii/S0140673611611434>
- [15] C. B. Traxler, “The Stanford Achievement Test, 9th Edition: National Norming and Performance Standards for Deaf and Hard-of-Hearing Students,” *J. Deaf Stud. Deaf Educ.*, vol. 5, no. 4, pp. 337–348, Sep. 2000.
- [16] W. G. Kronenberger, D. B. Pisoni, S. C. Henning, B. G. Colson, and L. M. Hazzard, “Working memory training for children with cochlear implants:

- a pilot study.,” *J. Speech. Lang. Hear. Res.*, vol. 54, no. 4, pp. 1182–96, Aug. 2011.
- [17] P. Johnson, S. Cawthon, B. Fink, E. Wendel, and S. Schoffstall, “Trauma and Resilience Among Deaf Individuals,” *J. Deaf Stud. Deaf Educ.*, vol. 23, no.4, pp. 31–330, Oct. 2018. Available: <https://academic.oup.com/jdsde/article/23/4/317/5056447>
- [18] S. S. Stevens, H. Davis, and S. Louis, “Hearing, Its Psychology and Physiology,” *The journal of the Acoustical Society of America*, vol 77 pp 1628-1629, 1985. Available: [www.abdi-ecommerce10.com/asa/images/product/medium/o-88318-4265.pdf](http://www.abdi-ecommerce10.com/asa/images/product/medium/o-88318-4265.pdf)
- [19] Samour, Barber.b.k, Giacaman “Mental Health Atlas 2005 - World Health Organization. Department of Mental Health and Substance Abuse, World Health Organization, World Health Organization. Mental Health: Evidence and Research, World Health Organization. Mental Health Evidence and Research Team - Google Livros.” pp 10-11, 2005. [Accessed: 11-Jan-2019].
- [20] L. J. Byrnes, J. Sigafos, F. W. Rickards, and P. M. Brown, “Inclusion of Students Who Are Deaf or Hard of Hearing in Government Schools in New South Wales, Australia: Development and Implementation of a Policy,” *J. Deaf Stud. Deaf Educ.*, vol. 7, no. 3, pp. 244–257, Jun. 2002 Available: [https://www.jstor.org/stable/42658619?seq=1#metadata\\_info\\_tab\\_contents](https://www.jstor.org/stable/42658619?seq=1#metadata_info_tab_contents).
- [21] E. KEATING and G. MIRUS, “American Sign Language in virtual space: Interactions between deaf users of computer-mediated video communication and the impact of technology on language practices,” *Lang. Soc.*, vol. 32, no. 05, pp. 693–714, Nov. 2003 Available: <https://www.cambridge.org/core/journals/language-in-society/article/american-sign-language-in-virtual-space-interactions-between-deaf-users-of-computer-mediated-video-communication-and-the-impact-of-technology-on-language-practices/8904A1FE2FA2D0663BA0C87282DB9850>.
- [22] Robert Traynor, “Sign Language Translation Devices—Robert Traynor—

- Hearing International”, “Hearing Health and Technology Matters”, Feb 2017. Available: <https://hearinghealthmatters.org/hearinginternational/2017/sign-language-translation-devices/>
- [23] “MotionSavvy – UNI Tablet Review | Center on Technology and Disability (CTD).” [Online]. Available: <https://www.ctdinstitute.org/library/2017-08-25/motionsavvy---uni-tablet-review>. [Accessed: 22-Jan-2020].
- [24] A. D. Hashim and F. Alizadeh, “Kurdish Sign Language Recognition System,” *UKH J. Sci. Eng.*, vol. 2, no. 1, pp. 1–6, Jun. 2018 Available: <https://journals.ukh.edu.krd/index.php/ukhjse/article/view/33>.
- [25] W. Oh and W. Brent Lindquist, “Image thresholding by indicator kriging,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 7, pp. 590–602, Jul. 1999 Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.56.5516>.
- [26] Yongmin Kim, Shijun Sun, “Object segmentation with background extraction and moving boundary techniques, ”Jun.2000, University of Washington. Available: [patents.google.com/patent/US6731799B1/en](https://patents.google.com/patent/US6731799B1/en)
- [27] “Arduino Uno Rev3.” [Online]. Available: <https://store.Arduino.cc/Arduino-uno-rev3>. [Accessed: 24-Jan-2019].
- [28] “What is a RaspberryPi Pi?” [Online]. Available: <https://www.RaspberryPi.org/help/what-is-a-RaspberryPi-pi/>. [Accessed: 24-Jan-2019].
- [29] “What is a RaspberryPi Pi? | Opensource.com.” [Online]. Available: <https://opensource.com/resources/RaspberryPi-pi>. [Accessed: 24-Jan-2019].
- [30] “RaspberryPi Pi — Teach, Learn, and Make with RaspberryPi.” [Online]. Available: <https://www.RaspberryPi.org/>. [Accessed: 24-Jan-2019].
- [31] “Cloud Text-to-Speech:síntesedefala | Cloud Text-to-Speech API | Google Cloud.” [Online]. Available: <https://cloud.google.com/text-to-speech/>. [Accessed: 01-Feb-2019].
- [32] Sreejan Alapati, Shivraj Yeole “A Review on Applications of Flex Sensors.”

- [Online]. Available:[https://www.researchgate.net/publication/318850816\\_A\\_Review\\_on\\_Applications\\_of\\_Flex\\_Sensors](https://www.researchgate.net/publication/318850816_A_Review_on_Applications_of_Flex_Sensors). [Accessed: 10-Jan-2019].
- [33] P. Polygerinos, K. C. Galloway, S. Sanan, M. Herman, and C. J. Walsh, 2017. "EMG Controlled Soft Robotic Glove for Assistance During Activities of Daily Living."
- [34] P. H. Marchetti and M. Duarte, "Instrumentação em Eletromiografia." Pp 7-29, Pontifícia Universidade Católica do Rio Grande do Sul, 2009. Available: <https://scholar.google.es/citations?user=UoynbjMAAAAJ&hl=en>
- [35] C. I. De Luca, "The Use of Surface Electromyography in Biomechanics", "Journal of Applied Biomechanics" vol 13 pp 135-163 1997. Available: <https://www.studocu.com/en/document/sapienza-universita-di-roma/cultural-anthropology-i-a/other/the-use-of-surface-electromyography-in-biomechanics/1425493/view>
- [36] Y. Fang and X. Chen, "Design and Simulation of UART Serial Communication Module Based on VHDL," *2011 3rd International Workshop on Intelligent Systems and Applications*, Wuhan, 2011, pp. 1-4. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5873448&isnumber=5873244>
- [37] Peter Norvigs *et al* "Pure Python spellchecking based on Peter Norvig's" Available: <https://pypi.org/project/pyspellchecker/>. [Accessed: 12-Jun-2019]
- [38] "I2C communication" Available: <https://www.i2c-bus.org/>. [Accessed: 15-Jun-2019]

